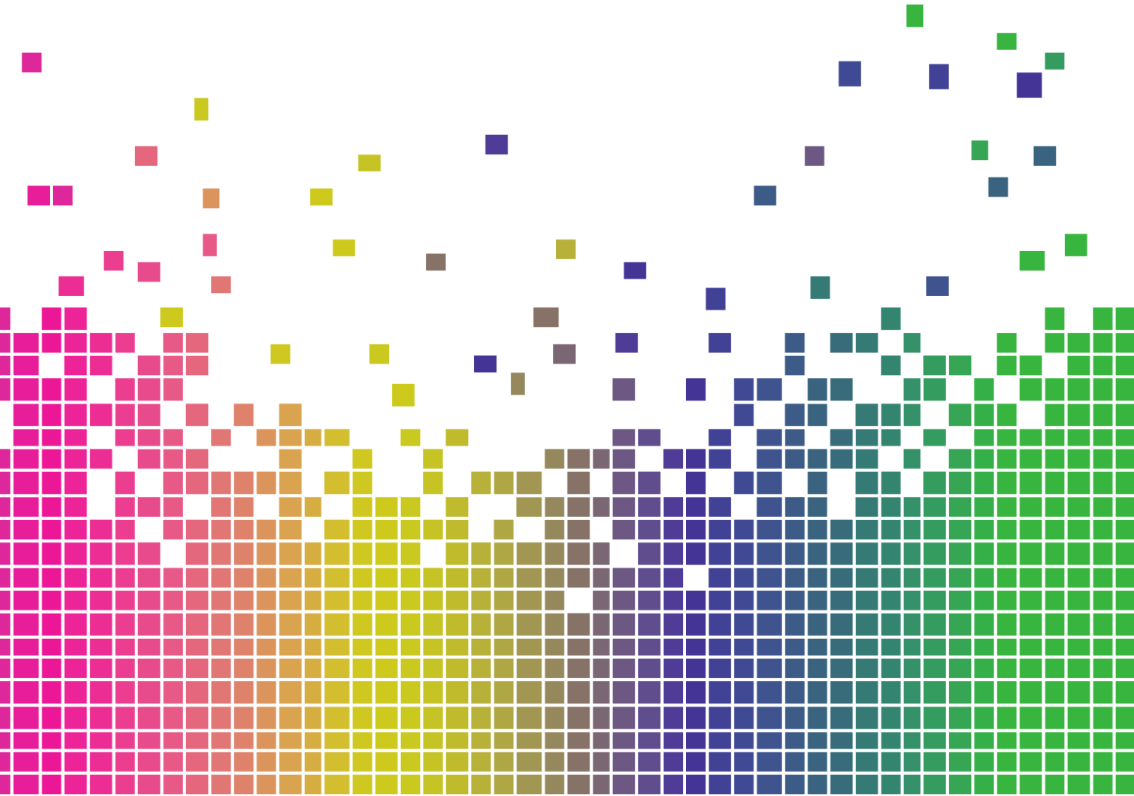
Actes de la conférence Traitement Automatique de la Langue Naturelle, TALN 2018

Vincent Claveau, Pascale Sébillot

Actes de la conférence TALN 2018

Volume 1 : Articles longs, articles courts de TALN

Pascale Sébillot, IRISA, INSA Rennes
Vincent Claveau, CNRS, IRISA, Univ. Rennes



Préface

Mots des présidents des comités de programme

Pour la première fois, l'ARIA (Association francophone de Recherche d'Information et Applications) et l'ATALA (Association pour le Traitement Automatique des Langues) ont organisé conjointement leur principale conférence annuelle afin de réunir en un seul lieu les deux communautés de la recherche d'information (RI) et du traitement automatique des langues (TAL). Organisée par l'IRISA (UMR 6074) et le Centre Inria Bretagne-Atlantique, cette édition s'est déroulée du 14 au 18 mai 2018 à Rennes. Elle a donc regroupé :

- la 15ème Conférence en Recherche d'Information et Applications (CORIA) ;
- la 25ème conférence sur le Traitement Automatique des Langues Naturelles (TALN) ;
- une rencontre jeunes chercheurs (RJC) commune aux deux communautés correspondant à la 13ème édition de la Rencontre des Jeunes Chercheurs en Recherche d'Information (RJCRI) et à la 20ème édition des Rencontre Étudiants Chercheurs en Informatique pour le Traitement Automatique des Langues (RECITAL) ;
- le salon de l'innovation en technologies du langage et de l'information.

Des ateliers et tutoriels, un hackathon ainsi qu'un salon de l'innovation à destination des industriels ont aussi enrichi ce programme (voir plus bas).

Les actes de CORIA ne sont pas présents dans ce volume mais sont accessibles à http://www.asso-aria.org/. Cette année, il y avait un seul format de soumission à CORIA mais deux formats de présentation pour les articles acceptés. Nous avons acceptés quinze articles pour une présentation longue et quatre articles pour des présentations courtes. Le taux de sélection pour les articles en présentation longue est de cinquante pourcent. Vingt-trois villes différentes sont représentées dans les dix-neuf papiers acceptés; beaucoup de travaux sont issus de collaborations, dont certaines internationales. Six papiers acceptés ont un auteur d'une organisation située à Toulouse, cinq d'une organisation parisienne et trois d'une organisation grenobloise. Au niveau international, nous pouvons noter des contributions acceptées provenant du Canada, de Russie et de Tunisie. Nous pouvons également noter des soumissions provenant du Cameroun et de Madagascar. La majorité des articles proviennent de laboratoires de recherche académiques. Les thèmes abordés à la fois dans les soumissions et dans les papiers acceptés sont variés tant au niveau des questions de recherche abordées que des méthodes proposées pour les résoudre et des collections utilisées pour valider ou évaluer les propositions.

Cette année, vingt deux articles ont été soumis à RJC. Après avoir été chacun évalué par trois membres du comité de programme, quatre articles ont été retenus pour une présentation orale (soit un taux de sélection pour présentation orale de 18 %), et neuf autres ont été retenus pour une présentation sous forme de poster (taux de sélection global de 59 %). Nous avons ainsi pu donner l'opportunité à treize jeunes chercheuses et chercheurs, en grande majorité en début de thèse, de présenter leurs travaux à la communauté.

Cette année, TALN inaugurait de nouvelles modalités de soumissions : un appel unique et un seul format de soumission en article court pouvant être étendu en article long sur proposition du comité de programme. Parmi les soixante douze articles soumis suite à cet appel, le comité de programme a proposé à quatorze d'entre eux un passage en format long (soit un taux de sélection de 19,5 %) et en a retenu quarante deux autres en articles courts. Pour effectuer cette sélection, le comité de programme s'est appuyé sur trois à quatre relectures effectuées par des membres du comité de lecture (liste donnée ci-après), synthétisées et portées lors de la réunion du comité de programme par les

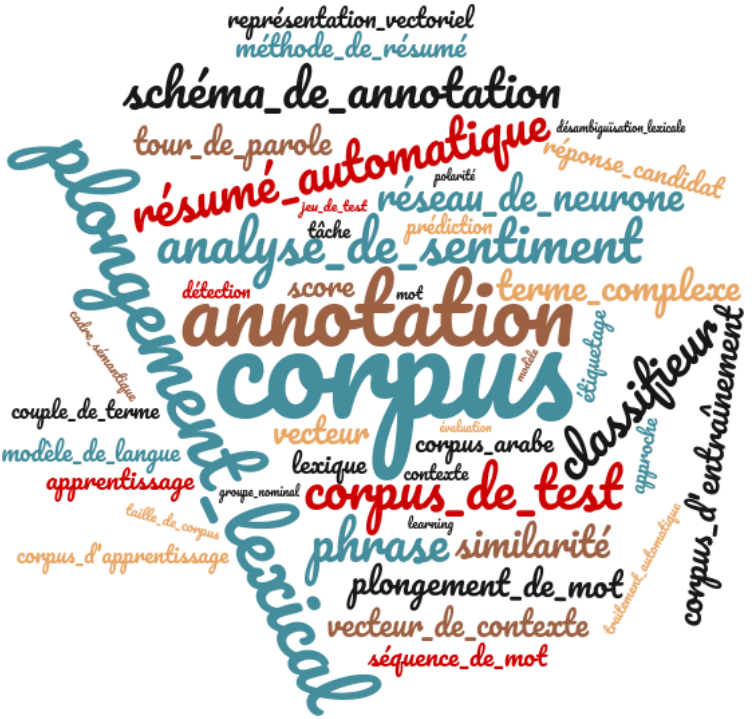


Figure 1: Nuage de termes extraits des actes de TALN.

responsables de domaine. L'ensemble de ce processus s'est déroulé comme les années précédentes en double aveugle. Les nombre de soumissions et le taux de sélection placent ainsi cette édition dans les pas de celles des années précédentes, suivant le double objectif d'avoir une conférence conservant d'une part une sélectivité forte, garante de la qualité des interventions orales, et se voulant, d'autre part, également un lieu de rencontre le plus ouvert possible à l'expression de l'ensemble de la communauté, au travers des articles courts.

Les thématiques abordées dans les articles retenus dans ces conférences sont variées. Sans surprise, les tendances de fond que constituent l'apprentissage profond et les plongements lexicaux occupent une part importante des contributions, mais pour autant d'autres approches et de nombreux domaines sont explorés. Les sessions ont ainsi porté sur les domaines d'application particuliers (domaines de spécialité, langues peu dotées), des niveaux d'analyses linguistiques (morphologie, syntaxe, lexique) ou des tâches spécifiques (résumé automatique, OCR, multimédia, fouille d'opinion). La figure 1 présente un nuage de termes extraits de ces actes[1].

En complément de ce programme, nous avons eu l'honneur d'accueillir deux oratrices invitées reconnues internationalement : Dina Demner (NUH, US National Library of Medicine) qui a présenté des avancées récentes en traitement automatique du langage biomédical, et Claudia Hauff (TU Delft) qui a effectué un exposé sur l'apprentissage humain en recherche d'information. Il convient également de citer le salon de l'innovation, qui, avec ses tables rondes, démonstrations, stands d'industriels du secteur ou de projets de recherche, permet aux industriels et aux chercheurs en TAL et RI, ainsi qu'aux entreprises en technologie de l'information et plus généralement du numérique, de se rencontrer

[1]Outils : TermEx, disponible sur https://allgo.inria.fr, et https://www.wordclouds.com.

et d'échanger autour des idées de développements actuels et futurs du domaine, de promouvoir les enjeux et applications du secteur, ainsi que de renforcer la visibilité et l'image des entreprises, organisations, institutions et projets de recherche auprès de partenaires et clients potentiels. Enfin, les conférences ont été précédées de deux journées d'ateliers et tutoriels se focalisant sur certaines thématiques plus précises du TAL et de la RI, portant sur la recherche d'information sémantique (atelier RISE), la fouille de texte (défi DeFT, cette année sur l'analyse de sentiment, dont les actes sont proposées dans le second volume du présent ouvrage), l'analyse des données de la recherche (atelier VaDOR), l'infrastructure de fouille de texte européenne OpenMinTed (tutoriel), le hackathon sur les fausse nouvelles ou infox (*fake news*), l'analyse des réseaux sociaux (atelier ALIAS, soutenu par le GdR CNRS MaDICS), et le data-journalisme (atelier CAJOLE, soutenu par le GdR CNRS MaDICS).

P. Cellier (RJC), A.-L. Ligozat (RJC), J. Mothe (CORIA), P. Sébillot (TALN), V. Claveau (TALN)

Mots des Présidents de l'ATALA et de l'ARIA

Cette année, les associations ARIA et ATALA ont souhaité organiser conjointement leur conférence à Rennes. L'objectif était de permettre aux chercheurs des deux communautés de se retrouver en un même lieu et un même temps autour de thématiques qu'ils partagent. En effet, le domaine de la Recherche d'Information ayant pour objectif d'identifier les informations les plus appropriées par rapport au besoin d'un usager, il repose sur différentes stratégies parmi lesquelles les modèles de langue trouvent une place spécifique. De même, dans le domaine du Traitement Automatique des Langues la transition du papier au support électronique nécessite des fonctionnalités se rapprochant de plus en plus des compétences humaines, phénomène amplifié par le retour sur le devant de la scène scientifique de l'Intelligence Artificielle, accompagné d'une demande croissante pour des agents informatiques donnant l'illusion de l'autonomie linguistique. Cette coïncidence n'est pas surprenante car la RI et le TAL partagent dès le début de l'informatique, une histoire commune avec l'IA, n'oublions pas en effet que les mesures d'évaluation emblématiques de la RI que sont la précision et le rappel ont été élaborées en 1960, lors des expériences du College of Aeronautics de Cranfield (UK) et qu'à la même époque, la communauté TAL se constituait autour de la traduction automatique, avec la naissance de l'ATALA à Paris, en 1959. Pour ce qui concerne l'IA, beaucoup considèrent l'atelier qui s'est tenu au Dartmouth College (USA) en 1956 comme marquant la naissance du domaine. C'est aussi pendant les années 60 que l'on a vu apparaître les premières implémentations d'algorithmes neuromimétiques pour l'apprentissage automatique. Nos deux communautés partageant des débuts contemporaines et étant unies comme par le passé autour de problématiques communes, nous avons donc fait le choix, cette année, de favoriser les échanges et les présentations communes au travers de l'organisation de ces conférences conjointes.

P. Paroubek (ATALA) & M. Chevalier (ARIA)

Comité d'organisation de CORIA-TALN-RJC

Coordinateur :

Vincent Claveau, CNRS, IRISA, Univ. Rennes

Webmestres :

Clément Dalloux, CNRS, IRISA, Univ. Rennes
Cédric Maigrot, IRISA, Univ Rennes

Resp. démonstrations :

Anne-Lyse Minard, CNRS, IRISA, Univ. Rennes

Resp. ateliers :

Annie Forêt, IRISA, Univ. Rennes

Resp. salon de l'innovation :

Géraldine Damnati, Orange, Lannion
Aleksandra Gerraz, Orange, Lannion

Resp. sponsoring :

Gwénolé Lecorvé, IRISA, ENSSAT, Univ. Rennes

Infographiste :

Agnès Cottais, IRISA, Rennes

Support administratif :

Élisabeth Lebret, Inria, Rennes
Aurélie Patier, IRISA, Rennes

Membres du comité d'organisation :

Cheikh Brahim El Vaigh, Inria, Rennes
Peggy Cellier, IRISA, INSA Rennes
Guillaume Gravier, IRISA, CNRS, Rennes
Pierre-François Marteau, IRISA, Univ. Bretagne Sud, Vannes
Nicolas Béchet, IRISA, IUT de Vannes
Pascale Sébillot, IRISA, INSA Rennes
Mikaïl Demirdelen, IRISA, INSA Rennes

Ainsi que les équipes techniques et administratives du centre Inria Rennes Bretagne Atlantique.

Comité de programme TALN

Présidents du comité de programme :

- Pascale Sébillot, IRISA, INSA Rennes
- Vincent Claveau, CNRS, IRISA, Univ. Rennes

Responsables de domaine :

Maxime Amblard, LORIA, Université de Lorraine
Delphine Bernhard, LiLPa, Université de Strasbourg
Philippe Blache, LPL, CNRS
Nathalie Camelin, LIUM, Université du Maine
Iris Eshkol-Taravella, MoDyCo, Université Paris Nanterre
Cécile Fabre, ERSS, Université Toulouse 2
Benoît Favre, LIF, Aix Marseille Université
Olivier Ferret, CEA LIST
Thierry Hamon, LIMSI, Université Paris Nord
Philippe Langlais, RALI/DIRO, Univ. de Montréal
Emmanuel Morin, LS2N, Université de Nantes
Philippe Muller, IRIT, Université Paul Sabatier
Aurélie Névéol, LIMSI, CNRS
Didier Schwab, LIG, Université Grenoble Alpes
Xavier Tannier, LIMICS, Université Pierre et Marie Curie

Comité de lecture :

Stergos Afantenos, IRIT, Université Paul Sabatier
Salah Ait-Mokhtar, NaverLabs
Alexandre Allauzen, LIMSI, Université Paris-Sud
Jean-Yves Antoine, LI, Université Tours
Frédéric Béchet, LIF, Aix Marseille Université
Laurent Besacier, LIG, Université Grenoble Alpes
Romaric Besançon, CEA LIST
Pierrette Bouillon, ETI/TIM/ISSCO, Université de Genève
Chloé Braud, LORIA, CNRS
Marie Candito, LLF, Université Paris Diderot
Thierry Charnois, LIPN, Université Paris 13
Chloé Clavel, Télécom Paris
Guillaume Cleuziou, LIFO, Université Orléans
Mathieu Constant, ATILF, Université Lorraine
Benoît Crabbé, LLf, Université Paris Diderot
Béatrice Daille, LS2N, Université Nantes
Laurence Danlos, LLF, Université Paris Diderot
Marco Dinarelli, LaTTiCe, CNRS
Patrick Drouin, RALI-DIRO, Université de Montréal
Thomas François, CENTAL, Université catholique de Louvain
Nathalie Friburger, LI, Université Tours
Claire Gardent, LORIA, CNRS
Éric Gaussier, LIG, Université Grenoble Alpes
Natalia Grabar, STL, CNRS
Camille Guinaudeau, LIMSI, Université Paris-Sud
Nabil Hathout, ERSS, Université de Toulouse
Nicolas Hernandez, LS2N, Université de Nantes
Stéphane Huet, LIA, Université d'Avignon et des pays de Vaucluse
Sylvain Kahane, Modyco, Université Paris Ouest - Nanterre
Olivier Kraif, LIDILEM, Université Grenoble Alpes
Mathieu Lafourcade, LIRMM, Université de Montpellier
Guy Lapalme, RALI-DIRO, Université de Montréal

Francois Lareau, OLST, Université de Montréal
Jean-Marc Lecarpentier, Greyc, Université Caen Basse-Normandie
Gwénolé Lecorvé, IRISA, ENSSAT, Université de Rennes
Joseph Le Roux, LIPN, Université Paris 13
Anaïs Lefeuvre-Halftermeyer, LIFO, Université d'Orléans
Sébastien Le Maguer,
Anne-Laure Ligozat, LIMSI, ENSIIE
Denis Maurel, LI, Université Tours
Anne-Lyse Minard, CNRS, IRISA, Université de Rennes
Richard Moot, LIRMM, CNRS
Véronique Moriceau, LIMSI, Université Paris-Sud
Adeline Nazarenko, LIPN, Université Paris Nord
Jian-Yun Nie, RALI-DIRO, Université de Montréal
Yannick Parmentier, LORIA, Université Lorraine
Sylvain Pogodalla, LORIA, INRIA
Thierry Poibeau, LaTTiCe, CNRS
Andrei Popescu-Belis, HEIG VD - School of Business and Engineering Vaud
Jean-Philippe Prost, LIRMM, Université Montpellier 2
Solen Quiniou, LS2N, Université Nantes
Christian Raymond, IRISA, INSA Rennes
Christian Retoré, LIRMM, Université de Montpellier
Mathieu Roche, CIRAD
Sophie Rosset, LIMSI, CNRS
Michel Simard, National Research Council Canada (NRC)
Ludovic Tanguy, ERSS, Université Toulouse 2
Isabelle Tellier, Lattice, Université Paris 3
Juan-Manuel Torres-Moreno, LIA, Université d'Avignon et des pays de Vaucluse
Julien Velcin, ERIC, Université Lyon 2
Guillaume Wisniewski, LIMSI, Université Paris-Sud
François Yvon, LIMSI, Université Paris-Sud
Pierre Zweigenbaum, LIMSI, CNRS

Table des matières

Articles longs

Articles courts

Articles longs

Étude de la lisibilité des documents de santé avec des méthodes d'oculométrie

Natalia Grabar[1] Emmanuel Farce[3] Laurent Sparrow[3]
(1) CNRS, UMR 8163 - STL - Savoirs Textes Langage, Univ. Lille, F-59000 Lille, France
(2) Univ. Lille, CNRS, UMR 9193 - SCALab, F-59000 Lille, France
natalia.grabar@univ-lille3.fr, emmanuel.farce@univ-lille3.fr,
laurent.sparrow@univ-lille3.fr

RÉSUMÉ
Le domaine médical fait partie de la vie quotidienne pour des raisons de santé, mais la disponibilité des informations médicales ne garantit pas leur compréhension correcte par les patients. Plusieurs études ont démontré qu'il existe une difficulté réelle dans la compréhension de contenus médicaux par les patients. Nous proposons d'exploiter les méthodes d'oculométrie pour étudier ces questions et pour détecter quelles unités linguistiques posent des difficultés de compréhension. Pour cela, des textes médicaux en version originale et simplifiée sont exploités. L'oculométrie permet de suivre le regard des participants de l'étude et de révéler les indicateurs de lecture, comme la durée des fixations, les régressions et les saccades. Les résultats indiquent qu'il existe une différence statistiquement significative lors de la lecture des versions originales et simplifiées des documents de santé testés.

ABSTRACT
Study of readability of health documents with eye-tracking methods.

Medical area is integral part of our lives due to health concerns, but the availability of medical information does not guarantee its correct understanding by patients. Several studies addressed this issue and pointed out real difficulty in understanding of health contents by patients. We propose to use eye-tracking methods for studying further the issue and for detecting which linguistic units in health documents are problematic. For this, original and simplified versions of medical documents are exploited. Eye-tracking permits to follow the gaze of participants and to reveal reading indicators such as duration of fixations, regressions and saccades. The results indicate that there is statistically significant difference in reading of original and simplified versions of the health documents tested.

MOTS-CLÉS : Lisibilité des documents, compréhension, alphabétisation médicale, oculométrie.

KEYWORDS: Readability of documents, understanding, health literacy, eye-tracking.

1 Introduction

Le domaine médical est de plus en plus présent dans la vie quotidienne des citoyens, essentiellement pour des raisons de santé et de soins médicaux mais également parce que ce type d'information peut être rencontré dans les médias ou la littérature. Avec l'évolution de l'Internet, les informations médicales deviennent aussi accessibles et disponibles en ligne. Il a été par exemple noté que l'Internet est devenu la deuxième source d'information pour les patients, en se plaçant après les consultations chez les médecins (Pletneva *et al.*, 2011; Fox, 2011). Ainsi, jusqu'à 24 % de la population utilise

l'Internet au moins une fois par jour pour trouver des informations relatives à leur santé et, de manière plus générale, jusqu'à 80 % de la population recourt à l'Internet pour trouver ce type d'informations. Néanmoins, la disponibilité des informations de santé ne présume pas qu'elles soient compréhensibles et correctement utilisées par les patients. Comme tous les domaines de spécialité, le domaine médical utilise des termes qui véhiculent des notions complexes, comme par exemple *tensosynovite, arthralgies* ou *métatarsophalangien*. Cette situation a été observée dans plusieurs études qui démontrent une faible compréhension des informations médicales par les citoyens (Patel *et al.*, 2002; Williams *et al.*, 1995; Berland *et al.*, 2001) et une communication compliquée entre les patients et les médecins (Jucks & Bromme, 2007; Tran *et al.*, 2009).

En relation avec ces questions, la complexité et la compréhension des textes et des termes sont étudiées par les chercheurs de différentes disciplines. Par exemple, la linguistique étudie la complexité morphologique des lexèmes (Iacobini, 2003; Lüdeling *et al.*, 2002), la psychologie étudie différents facteurs internes et externes aux mots qui influencent leur reconnaissance par les locuteurs (Bertram *et al.*, 2011; Lüttmann *et al.*, 2011; Bozic *et al.*, 2007; Dohmes *et al.*, 2004; Cain *et al.*, 2009), la terminologie étudie comment différencier les bons candidats-termes, qui véhiculent le sens bien spécialisé, au sein d'une extraction effectuée automatiquement (Kageura & Umino, 1996; Frantzi *et al.*, 2000; Hamon *et al.*, 2014), et le TAL propose des méthodes automatiques pour faire la différenciation entre les textes ou termes difficiles et ceux de la langue générale (Zeng *et al.*, 2005; Chmielik & Grabar, 2011; Shardlow, 2013).

L'objectif de notre travail consiste à étudier la compréhension des informations de santé par les non-experts. Plus particulièrement, nous proposons de voir quel est l'impact de la simplification des termes. Nous travaillons avec des données en français. Nous exploitons pour ceci les méthodes et les outils fournis par la psychologie, et plus particulièrement l'oculométrie (*eye-tracking*). En effet, l'étude des mouvements oculaires pendant la lecture fournit des indications objectives et précieuses sur les processus cognitifs impliqués. Plus particulièrement, la difficulté et la lisibilité d'un texte peuvent être mesurées avec différents indicateurs (Duchowski, 2007; Rayner, 1998; Sparrow *et al.*, 2003; Miellet *et al.*, 2008). Il s'agit principalement des indicateurs suivants :

— Les *saccades oculaires* sont des mouvements rapides des yeux pour aller d'un point de lecture vers un autre. Lorsque le texte est facile à lire et à assimiler, les saccades sont longues, alors qu'elles deviennent plus courtes avec un texte compliqué ;
— Les *fixations* sont des périodes pendant lesquelles les yeux sont stables. Les fixations correspondent aux moments lorsque l'information visuelle est analysée. La durée des fixations augmente avec des textes difficiles car ils nécessitent un temps d'assimilation plus important ;
— Les *régressions* sont les retours en arrière vers les endroits du texte déjà lus par le lecteur. La lecture d'un texte difficile engendre en général plus de régressions.

Ainsi, la comparaison de paramètres oculomoteurs (durée des fixations, amplitude des saccades, régressions, etc.) enregistrés pendant la lecture des textes va permettre d'évaluer plus précisément les difficultés et les points de blocage de la part des lecteurs. Selon notre hypothèse, la lecture de textes complexes et de termes inconnus conditionne notre attention et les mouvements oculaires présentent alors des comportements typiques et observables. De tels indices peuvent donc être directement corrélés avec les difficultés de compréhension.

Les indicateurs d'oculométrie sont exploités dans différents domaines. En psychologie et en neurosciences, ils permettent de collecter les informations sur le fonctionnement du cerveau, les modalités de la lecture et de l'attention, etc. (Cooper, 1974; Molnar, 1981; Asaad & andEK Miller, 2000; Rayner & Liversedge, 2004; Clifton *et al.*, 2007). En marketing et en publicité, ils fournissent les indicateurs qui permettent d'améliorer la présentation des informations, de mettre les bonnes informations aux

bons endroits ou de comprendre à quels types d'informations les utilisateurs et acheteurs sont les plus sensibles (Andrews & Coppola, 1999; Higgins *et al.*, 2014). Dans des contextes liés à l'acquisition et au traitement de la langue, les indicateurs d'oculométrie sont typiquement exploités pour détecter les endroits dans les textes qui attirent ou bloquent le regard ou la lecture, comme par exemple : la lecture de textes en langues maternelle et secondaire (Altarriba *et al.*, 1996; Bisson *et al.*, 2014) ; la lecture de textes par les personnes dyslexiques (Rubino & Minden, 1973; Elterman *et al.*, 1980; Rello *et al.*, 2013; Nilsson Benfatto *et al.*, 2016) ou autistes (Yaneva *et al.*, 2015) ; le traitement de structures syntaxiques (Frenck-Mestre & Pynte, 1997; Clifton & Staub, 2011; Trueswell *et al.*, 1994; Singh *et al.*, 2016) ; la relation entre le traitement de la parole et les mouvements oculaires, où les participants montrent la tendance de fixer l'image correspondant à la phrase qu'ils entendent à l'oral (Cooper, 1974; Tanenhaus *et al.*, 1995; Wendt *et al.*, 2014) ; la détection et le traitement des erreurs (Keating, 2009) ; l'évaluation de la complexité des textes lors de la traduction (Sharmin *et al.*, 2008) ou de l'acquisition de langues (Balakrishna, 2015). Le plus souvent, c'est l'information sur les fixations qui est exploitée. Par exemple, en relation avec le marketing et la publicité, la figure 1 montre une page de résultats d'un moteur de recherche en ligne. Les couleurs chaudes indiquent les endroits qui attirent le plus le regard des utilisateurs : les premiers résultats de la recherche et, en moindre mesure, la publicité sur le coté. La zone la plus fixée forme ce qu'on appelle le *golden triangle*. Ce modèle indique aussi que l'attention des utilisateurs décroit assez rapidement.

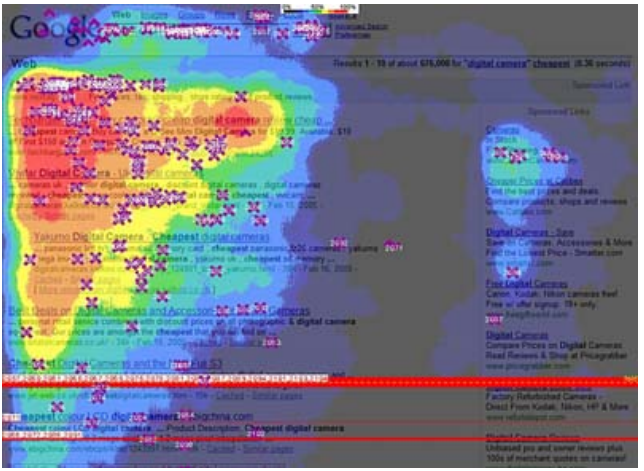


FIGURE 1 – Le modèle de fixations des pages de résultats d'un moteur de recherche.

2 Approche proposée

Notre approche est basée sur l'exploitation de méthodes d'oculométrie pour étudier les modalités de lecture de textes médicaux. Nous décrivons d'abord les documents utilisés et les critères d'inclusion des participants. Nous présentons ensuite le déroulement des tests et l'analyse des données obtenues.

EXAMEN : ECHOGRAPHIE DES MAINS ET DES PIEDS
MOTIF : Bilan d'arthralgies
Mains : On ne visualise pas de ténosynovite, ou d'arthrosynovite.
Avant-pieds : On retrouve des remaniements intéressant les premières métatarsophalangiennes en rapport avec des antécédents de chirurgie d'hallux valgus.
Absence d'arthrosynovite au niveau des articulations métatarsophalangiennes.

EXAMEN : ECHOGRAPHIE DES MAINS ET DES PIEDS
MOTIF : Bilan de douleurs articulaires
Mains : On ne visualise pas d'inflammation des tendons, ni de la membrane articulaire.
Avant-pieds : On retrouve des remaniements intéressants sur les premières articulations des pieds en rapport avec les antécédents de la chirurgie de la déformation du pied.
Absence d'inflammation de la membrane au niveau des articulations du pied.

FIGURE 2 – Texte$_1$ en version originale (en haut) et simplifiée (en bas) lu à l'étape 3 de la passation.

2.1 Documents cliniques et leur préparation

Deux extraits de documents cliniques sont utilisés, provenant d'un rapport de radiologie (figure 2) et d'une lettre de sortie en cardiologie (figure 3). Ces extraits sont exploités en deux versions : une version originale (technique) et une version simplifiée manuellement par un chercheur en TAL. La simplification est effectuée essentiellement au niveau lexical en utilisant les paraphrases de termes comme {*arthralgie*, *douleur articulaire*}, {*ténosynovite*, *inflammation d'un tendon*} ou {*infarctus du myocarde*, *crise cardiaque*} en partie issues de travaux antérieurs (Grabar & Hamon, 2016; Antoine & Grabar, 2016), ou bien des hyperonymes de termes {*métatarsophalangien→pied*}. Dans le texte$_1$, 7 changements lexicaux sont effectués qui correspondent aux substitutions de quatre composés néoclassiques (*arthralgie, ténosynovite, arthrosynovite, métatarsophalangien*) et d'un emprunt latin ({*hallux valgus→déformation du pied*}). Dans le texte$_2$, 10 changements sont effectués : substitutions par des équivalents ({*infarctus du myocarde*, *crise cardiaque*}, {*entre 35 et 40 %*, *basse*}), substitutions par des hyperonymes ({*angioplastie→intervention chirurgicale*}, {*endoprothèse→un autre stent*}, {*IVA moyenne→artère cardiaque*}, {*circonflexe proximale→une autre artère*}), et suppressions (*antérieur, microcytaire* et informations techniques sur les stents).

Les figures 2 et 3 présentent les versions originales et simplifiées des texte$_1$ et texte$_2$, respectivement. Comme nous pouvons le voir, les versions originales et simplifiées restent équivalentes au niveau du sens. Il s'agit de textes courts (48 et 65 mots pour le texte$_1$ et 112 et 82 mots pour le texte$_2$, version originales et simplifiées, respectivement) afin qu'ils puissent être facilement visualisés et lus sur un écran d'ordinateur. Ces textes sont exploités pour constituer deux ensembles de test :

1. texte$_1$ original et texte$_2$ simplifié,
2. texte$_1$ simplifié et texte$_2$ original.

2.2 Critères d'inclusion

Cinquante participants sont recrutés et chaque ensemble de test est lu par 25 participants, afin que la significativité statistique des paramètres de la lecture puisse être calculée entre ces deux versions des textes, tout en prenant en compte les éventuels problèmes techniques et de calibrage.

Cette patiente avait constitué un infarctus du myocarde antérieur en novembre 2010, pour lequel avait été réalisée une angioplastie de l'IVA moyenne avec implantation d'un stent non actif Vision de 2.75 mm x 18 mm, un complément par angioplastie au ballon seul en aval. Une endoprothèse avait également été implantée au niveau de la circonflexe proximale, avec un stent Vision 2.5 x 18 mm. La fraction d'éjection était évaluée entre 35 et 40 %.
Nous l'avions revue récemment, en insuffisance cardiaque, avec plusieurs autres problèmes :
- une anémie microcytaire inexpliquée,
- un déséquilibre important de son diabète pour lequel elle a été, entre temps, prise en charge par nos confrères diabétologues.

Cette patiente avait présenté une crise cardiaque en novembre 2010, pour laquelle avait été réalisée une intervention chirurgicale de l'artère cardiaque avec implantation d'un stent non actif. Un autre stent avait également été implanté au niveau d'une autre artère. La fraction d'éjection observée était basse.
Nous l'avions revue récemment, en insuffisance cardiaque, avec plusieurs autres problèmes :
- une anémie inexpliquée,
- un déséquilibre important de son diabète pour lequel elle a été, entre temps, prise en charge par nos confrères diabétologues.

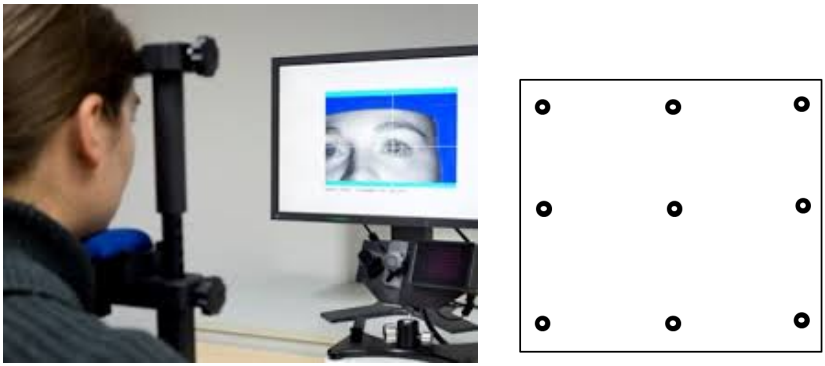
FIGURE 3 – Texte$_2$ en version originale (en haut) et simplifiée (en bas) lu à l'étape 3 de la passation.

Peuvent être inclus dans l'étude :

— les étudiants de L1 (niveau licence 1^{ere} année) de différentes disciplines (psychologie, linguistique, arts plastiques...). Il s'agit d'étudiants venant de terminer leurs études secondaires et ayant suivi les premiers mois d'étude à l'université. Les étudiants poursuivant des études médicales et paramédicales sont exclus. En général, on distingue 5 niveaux d'alphabétisation (Bernèche & Perron, 2006), où les niveaux 1 et 2 correspondent aux personnes qui ont des difficultés de lecture et d'assimilation des informations, alors que les niveaux 4 et 5 fournissent la capacité de faire des déductions complexes, ce qui est souvent propre aux personnes ayant suivi des études supérieures. À notre avis, les étudiants L1 ont le plus souvent le niveau 3 d'alphabétisation : ils savent lire et écrire, ils peuvent assimiler de nouvelles informations et connaissances, mais n'ont pas de connaissances spécialisées y compris dans le domaine médical. Nous considérons ainsi que cette population peut représenter le citoyen lambda ;
— sans pathologies chroniques car cela peut impliquer une familiarité avec le domaine médical ;
— de langue maternelle française.

2.3 Déroulement des tests

L'approche proposée est basée sur l'exploitation d'oculométrie, qui permet d'évaluer la fluidité de la lecture avec des mesures objectives (Sparrow *et al.*, 2003; Miellet *et al.*, 2008). Comme indiqué plus haut, ces indicateurs permettent de détecter les zones de texte qui bloquent la lecture



(a) Caméra d'oculométrie (b) Calibrage à 9 points

FIGURE 4 – Le caméra d'oculométrie et son calibrage avec 9 points de fixation (étape 1).

et la compréhension. Les textes sont présentés sur un écran d'ordinateur et une caméra spécifique enregistre les mouvements des yeux qui peuvent ensuite être superposés sur le texte.

Lors des passations de chaque participant, plusieurs étapes sont effectuées. Après la présentation des objectifs de l'étude et la collecte du formulaire de consentement, chaque participant effectue :

1. Le calibrage du caméra d'oculométrie (modèle EyeLink 1000, comme celui de la figure 4(a)). Une fois le participant installé et sa pupille repérée par le caméra, il est nécessaire d'entraîner le logiciel d'oculométrie aux mouvements des yeux de ce participant. Le calibrage à 9 points est alors effectué (figure 4(b)). Chaque point apparaît dans un ordre aléatoire sur l'écran et le participant est demandé de ne pas anticiper la localisation des apparitions mais de les suivre. Une première passe permet au logiciel d'apprendre les mouvements et positions oculaires alors que la deuxième passe permet de les précalculer, de les comparer avec les mouvements et positions réels, et de calculer ensuite la différence prédictible entre les deux positions ;
2. La lecture d'un texte général pour l'entraînement du participant. Il s'agit d'un texte pour enfant qui parle d'une petite fille qui habite dans la forêt. L'objectif de cette étape est d'entraîner le participant à la tâche de lecture et de réponse aux questions ;
3. La lecture des deux textes médicaux (figures 2 et 3), la version originale et la version simplifiée, comme décrit dans la section 2.1 ;
4. La lecture d'un texte de contrôle (figure 5) avec le contenu médical grand public qui concerne la cardiologie. L'objectif de cette étape est d'avoir une référence de lecture d'un texte médical grand public pour vérifier d'une autre manière l'objectivité des indicateurs d'oculométrie. On s'attend ici que les indicateurs de lecture soient stables avec les participants ;
5. Après la lecture de chaque texte, le participant répond à un QCM (questionnaire à choix multiples) pour évaluer sa compréhension. Par exemple, sur le texte$_2$, une des questions posées est *La patiente présente les problèmes : 1) cardiaques, 2) cérébraux, 3) je ne sais pas*.

À la fin, si souhaité, les résultats obtenus avec un participant lui sont présentés et interprétés. Au total, la passation d'un participant dure entre 15 et 20 minutes. Les passations sont effectuées par un chercheur en TAL ou par un psychologue, en respectant les mêmes conditions expérimentales de laboratoire et sans distracteur.

Le cœur est irrigué par les artères coronaires, qui sont elles-mêmes alimentées par une autre grande artère : l'aorte. Quand le diamètre des coronaires se rétrécit à cause de la formation progressive de plaques de graisse, le muscle cardiaque n'est plus assez alimenté en oxygène et en nutriments : il est en souffrance. Si l'artère se bouche complètement, l'infarctus guette… Les techniques du pontage ou de la pose d'un stent ont le même but : rétablir une circulation sanguine normale.

FIGURE 5 – Texte de contrôle (étape 4).

2.4 Analyse des données

Les données collectées pendant les tests d'oculométrie sont analysées avec le *test t* pour calculer la significativité que nous observons au niveau des indicateurs de lecture des textes originaux et simplifiés, des textes d'entraînement (étape 2) et de contrôle (étape 4), et des réponses aux QCM.

Les données d'oculométrie sont aussi comparées avec les résultats d'annotation manuelle effectuée sur des données similaires. Il a été ainsi demandé aux annotateurs de marquer les segments de textes qu'ils ne peuvent pas comprendre. Nous supposons que les deux approches peuvent indiquer des unités linguistiques similaires, qui sont difficiles à comprendre, et que, contrairement aux annotations, dont le résultat correspond à des choix conscients, l'enregistrement des mouvements oculaires lors de la lecture fournit des indices non conscients de la part des participants. Il s'agit en effet des habitudes individuelles de lecture acquises lors de l'apprentissage scolaire et familial.

3 Résultats et discussion

	Texte$_1$						*Texte*$_2$					
	O	*S*	*écart*	*p*	*ddl*	*t-test*	*O*	*S*	*écart*	*p*	*ddl*	*t-test*
TRN	60,55	63,63	-3,08	0,23	45,00	1,22	62,73	59,67	3,06	0,22	45,00	1,24
CRL	58,88	62,06	-3,19	0,22	45,00	1,25	61,04	57,84	3,20	0,21	45,00	1,29
DPF	227,41	215,75	11,66	0,11	45,00	1,65	214,73	214,69	0,04	0,50	45,00	0,68
NTF	587,61	370,48	217,14	**0,00**	45,00	7,38	395,71	372,22	23,49	0,16	45,00	1,43
AMP	3,50	3,80	-0,30	**0,02**	45,00	2,44	3,33	3,82	-0,49	**0,00**	45,00	5,38
REG	27,26	21,21	6,06	**0,05**	45,00	2,05	21,47	19,30	2,18	0,24	45,00	1,18
QCM	1304,35	869,57	434,78	**0,02**	21,00	2,08	602,77	538,95	63,82	**0,00**	21,00	2,08

TABLE 1 – Résultats des passations pour les deux textes en versions originale O et simplifiée S et leur analyse statistique. Les indicateurs traités sont : les indicateurs pour les textes d'entraînement *TRN* et de contrôle *CRL* ; la durée de la première fixation *DPF*, le nombre total de fixations *NTF*, l'amplitude des saccades *AMP*, le nombre de régressions *REG* ; les réponses aux questions *QCM*. Le p statistiquement significatif est marqué en gras.

Le tableau 1 présente les résultats des indicateurs obtenus et leur analyse statistique pour le texte$_1$ et le texte$_2$. L'analyse indique les valeurs suivantes : les moyennes pour les versions originales et simplifiées, l'écart-type, le p, le degré de liberté ddl et le *test t*. Les figures 6 et 7 montrent les exemples de lecture des versions originales (en haut) et simplifiées (en bas) du texte$_1$ et du texte$_2$,

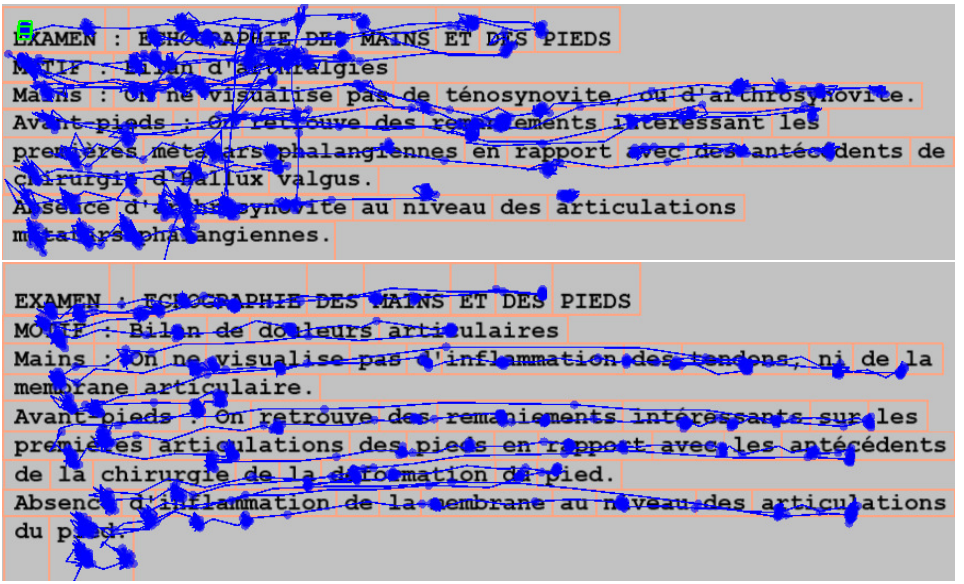


FIGURE 6 – Exemple de lecture du texte$_1$ en version originale (en haut) et simplifiée (en bas).

respectivement. Nous présentons et discutons les différents indicateurs obtenus :

— *Lecture des textes d'entraînement (TRN) et de contrôle (CRL).* Entre les deux ensembles de tests, il n'existe pas de différence statistique dans les paramètres étudiés lors de la lecture des textes d'entraînement (*TRN*) et de contrôle (*CRL*). Cela indique que les participants ont la même capacité de lecture de manière générale : ainsi, les indicateurs collectés lors de la lecture des textes médicaux sont comparables. Cet indicateur donne d'autant plus de valeurs aux résultats obtenus avec la lecture des textes médicaux originaux et simplifiés ;

— *Durée de la première fixation (DPF).* Pour les texte$_1$ et texte$_2$, il n'existe pas de différence de la durée de la première fixation (*DPF*) entre les deux versions de ces textes. Cela indique que la lecture de ces textes commence de la même manière et sans anticipation de la part du participant. Ceci est aussi un bon indicateur qui renforce d'autres résultats car il montre que les résultats ne seront pas biaisés par l'attente ou le comportement des participants ;

— *Nombre total de fixations (NTF).* Pour le texte$_1$, une différence statistiquement significative est observée en relation avec le nombre total de fixations (*NTF*) : sur la version originale du texte$_1$, les fixations sont plus longues et nombreuses. Cela peut être observé sur la figure 6 : sur le texte original (en haut), les points bleus des fixations sont plus grands par comparaison avec le texte simplifié (en bas). Par exemple, dans le texte original, les mots comme *arthralgie* ou *métatarsophalangiennes* montent de longues fixations. Certains mots du texte$_1$ (*arthrosynovite, arthralgie, métatarsophalangiennes*) peuvent avoir plusieurs points de fixation. En revanche, le texte$_2$ ne montre pas de différence statistique par rapport au nombre total de fixations ;

— *Amplitude des saccades (AMP).* Avec les texte$_1$ et texte$_2$, la simplification mène à une augmentation statistiquement significative des amplitudes des saccades (*AMP*). Cela signifie que la simplification diminue effectivement la difficulté de lecture et que l'oeil peut faire des sauts plus grands dans le texte pour aller d'un point de lecture vers un autre. Cela peut être observé sur les figures 6 et 7 : sur le texte original (en haut), les lignes bleues horizontales

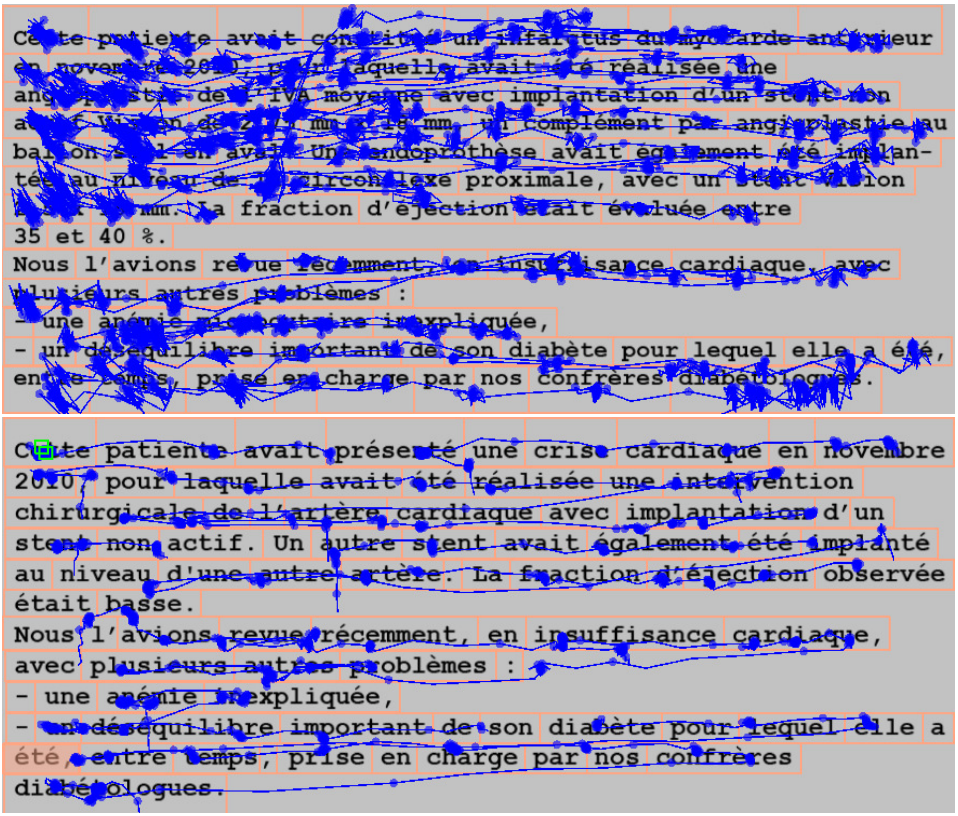


FIGURE 7 – Exemple de lecture du texte$_2$ en version originale (en haut) et simplifiée (en bas).

sont plus courtes que sur le texte simplifié ;

— *Régressions (REG).* Pour le texte$_1$, le nombre de régressions (*REG*) est significativement moins important lors de la lecture des textes simplifiés, ce qui suggère que la lecture et la compréhension de la version simplifiée sont meilleures. Cela peut aussi être observé sur la figure 6 : sur le texte original, nous pouvons voir les lignes bleues verticales. Pour le texte$_2$, cet indicateur ne montre pas de différence statiquement significative ;

— *Réponses des QCM.* L'analyse des réponses aux QCM indique que la compréhension des versions simplifiées est meilleure pour les texte$_1$ et texte$_2$: nous obtenons 54 % et 50 % de réponses correctes avec les textes originaux, alors que les textes simplifiés fournissent 85 % et 78 % de réponses correctes, respectivement. Cette différence est aussi statistiquement significative pour les deux textes ;

— *Comparaison entre les fixations et les annotations manuelles.* La comparaison des fixations avec les annotations manuelles indique que les lecteurs se focalisent plus longtemps sur les types de termes et d'informations techniques qui peuvent être marqués lors de l'annotation manuelle du même type de textes. Il s'agit typiquement des composés néoclassiques (*arthralgie*, *métatarsophalangienne*, *arthrosynovite*), des termes anatomiques (*métatarsophalangienne*), des dérivés avec un sens spécialisé (*antécédent, éjection, anémie*). Dans le texte$_2$, il y a également les valeurs numériques qui bloquent la lecture. Notons que les valeurs numériques

peuvent aussi être marquées lors de l'annotation manuelle des textes. De manière surprenante, l'emprunt latin *Hallux valgus* du texte$_2$ n'a pas bloqué le regard lors de la lecture. Pour certains participants, cet emprunt est associé avec un nom propre à cause de la majuscule au début : sa lecture n'a donc pas sollicité de fixation. Dans les exemples (1)-(3), nous montrons les résultats de l'annotation manuelle d'un même segment de texte par trois participants. Les passages en rouge sont marqués comme ne pouvant pas être compris. Nous pouvons clairement voir qu'il existe une différence importante dans les annotations. Chaque participant suit ses propres principes d'annotation : focalisation sur les mots ou sur les expressions plus complexes, annotation de la première occurrence ou de toutes les occurrences, une certaine familiarité ou une bonne connaissance des notions... De plus, l'annotation d'une unité donnée reste toujours un choix conscient avec ce type d'approche. Il s'agit donc d'une méthode plus subjective par rapport aux méthodes et indicateurs d'oculométrie. À notre avis, il s'agit d'un point très positif de l'oculométrie.

(1) *L'échographie cardiaque montre des cavités gauches dilatées (OG 59, VG 61/40, fraction d'éjection 61%, PAPS 38, bon fonctionnement de la Saint aortique N 25 (gradient moyen à 16.5), aspect satisfaisant du montage du tube valvulaire aortique. Sur la mitrale, il s'agit d'une mitrale dystrophique avec un prolapsus de P2 et éversion systolique complète dans l'oreillette gauche par rupture de cordage, calcification mitrale postérieure, fuite mitrale cotée 4/4. L'insuffisance tricuspidienne apparaît minime.*

(2) *L'échographie cardiaque montre des cavités gauches dilatées (OG 59, VG 61/40, fraction d'éjection 61%, PAPS 38, bon fonctionnement de la Saint aortique N 25 (gradient moyen à 16.5), aspect satisfaisant du montage du tube valvulaire aortique. Sur la mitrale, il s'agit d'une mitrale dystrophique avec un prolapsus de P2 et éversion systolique complète dans l'oreillette gauche par rupture de cordage, calcification mitrale postérieure, fuite mitrale cotée 4/4. L'insuffisance tricuspidienne apparaît minime.*

(3) *L'échographie cardiaque montre des cavités gauches dilatées (OG 59, VG 61/40, fraction d'éjection 61%, PAPS 38, bon fonctionnement de la Saint aortique N 25 (gradient moyen à 16.5), aspect satisfaisant du montage du tube valvulaire aortique. Sur la mitrale, il s'agit d'une mitrale dystrophique avec un prolapsus de P2 et éversion systolique complète dans l'oreillette gauche par rupture de cordage, calcification mitrale postérieure, fuite mitrale cotée 4/4. L'insuffisance tricuspidienne apparaît minime.*

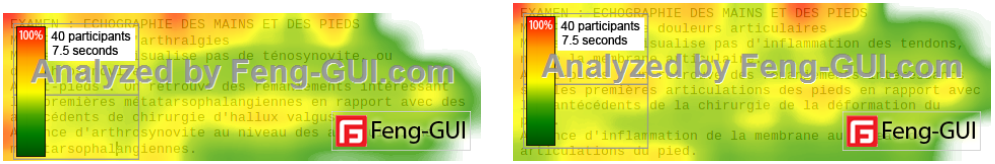
Si l'oculométrie offre plusieurs indicateurs objectifs sur le processus de lecture (typiquement le nombre et la durée des fixations, l'amplitude des saccades, le nombre de régressions) et la possibilité de les corréler avec d'autres informations comme la compréhension des textes, cette méthode présente aussi quelques limitations connues et prises en compte pendant les expériences (Duchowski, 2007) :

— Les appareils d'oculométrie permettent de capter et d'enregistrer le regard des participants. La supposition communément faite alors est que le regard coïncide avec l'attention du sujet, alors qu'en réalité l'attention peut aussi être portée sur un objet qui se trouve en périphérie du regard, par exemple. Le système de vision humaine est en effet très complexe et n'est pas complètement décodé pour le moment. Il s'agit d'une des limitations connues des méthodes d'oculométrie qui demande donc d'accepter les deux possibilités (le regard correspond à l'attention explicite ou non). Dans notre cas, avec la lecture de textes médicaux courts et le besoin de répondre aux questions, et à l'absence de distracteurs, nous pouvons supposer que

l'attention des lecteurs coïncide avec leur regard ;

— Avec certains participants, dû aux caractéristiques physiologiques (cils longs, maquillage, paupières lourdes...), il peut être difficile d'effectuer le calibrage du caméra et de bien capter et enregistrer les mouvements oculaires. Cela peut mener à une perte des données ou bien à une mauvaise superposition des enregistrements sur le texte. En revanche, lorsque les données sont exploitables, cela n'influence pas les indicateurs de lecture ;
— De la même manière, le port de lunettes et de lentilles de contact peut poser des difficultés lors de la capture de la pupille ;
— Avec un même texte ou image, l'attention et donc le regard des participants varient en fonction de la tâche ou des questions qui leur sont soumises. Dans notre cas, tous les participants devaient accomplir la même tâche qui consistait en lecture des textes et en réponse aux questions de compréhension ensuite ;
— Le matériel d'oculométrie a des limites aussi : (1) il fonctionne avec une certaine fréquence (60 Hz) et certains mouvements oculaires peuvent ne pas être captés et enregistrés ; (2) le signal enregistré est débruité, comme par exemple les clignotements ou certains mouvements périphériques ce qui peut aussi supprimer certains mouvements oculaires importants.

De manière générale, selon les résultats d'oculométrie, nous notons que la simplification des textes améliore les indicateurs de lecture : (1) la durée totale des fixations devient plus courte, (2) l'amplitude des saccades devient plus grande, (3) les régressions deviennent moins fréquentes, (4) la compréhension du contenu et les réponses aux questions s'améliorent. Tous ces indicateurs de vérifient sur le texte$_1$ et une partie de ces indicateurs se vérifie sur le texte$_2$. Ceci est également un résultat positif du travail. Ces résultats fournissent des indicateurs cohérents obtenus suite à la lecture des textes techniques en versions originales et simplifiées. Ils indiquent qu'il existe des schémas spécifiques lors de la lecture des textes techniques où le regard est attiré par certains éléments langagiers, comme les composés néoclassiques, les termes anatomiques ou les valeurs numériques.



(a) Texte$_1$ original (b) Texte$_1$ simplifié

FIGURE 8 – La simulation des fixations pour le texte$_1$.

Notons qu'il est possible de faire des simulations d'oculométrie sur certains sites en ligne. Pour ceci, il faut soumettre une image du texte et la simulation des fixations est obtenue très rapidement. La figure 8 montre des exemples de telles simulations sur le texte$_1$ pour la version originale (figure 8(a)) et la version simplifiée (figure 8(b)). Les couleurs chaudes représentent les endroits qui attirent plus le regard. Il nous semble que cette simulation est plutôt construite selon le modèle de fixations de la figure 1, où le début du texte et les débuts des lignes attirent le plus le regard de l'utilisateur. Comme nous pouvons le voir, ce modèle de lecture est très différent de celui que nous avons obtenu avec la lecture de textes techniques (figures 6 et 7). Il s'agit d'un résultat intéressant car il montre que différents types de textes peuvent solliciter différents types d'attention et de lecture.

4 Conclusion

Nous avons proposé une expérience pour étudier l'effet de la simplification de textes techniques, sur l'exemple de textes médicaux, grâce aux méthodes d'oculométrie. De cette manière, nous obtenons plusieurs indicateurs de lecture objectifs, comme la durée des fixations, la durée de la première fixation, l'amplitude des saccades et les régressions. Ces mêmes indicateurs sont obtenus suite à la lecture d'un texte d'entraînement et d'un texte de contrôle. Les indicateurs collectés sont comparés entre les versions originales et simplifiées d'un texte médical donné avec des mesures statistiques (*test t*) pour analyser s'il existe une différence statistiquement significative lors de la lecture des versions originales et simplifiées de ces textes. Ensuite, nous analysons les réponses aux questionnaires QCM fournies par les participants après la lecture de chaque texte et les annotations manuelles des difficultés de compréhension.

Les résultats obtenus indiquent que la lecture des deux versions des textes, originales et simplifiées, fournit des indicateurs cohérents et stables. Par exemple, lors de la lecture des textes simplifiés, les fixations sont plus courtes, les saccades plus longues et les régressions absentes ou pas fréquentes. De plus, l'analyse des réponses aux questions indique que la compréhension des textes simplifiés est meilleure : ainsi, sur le texte$_1$, le nombre de réponses correctes est de 54 % pour la version originale et 81 % pour la version simplifiée. Cela indique aussi que les textes techniques, et en particulier les textes médicaux, peuvent être simplifiés de manière efficace pour atteindre une meilleure compréhension de la part des non-experts. Notons aussi que la comparaison avec l'annotation manuelle, effectuée par le même type de participants, montre que les lecteurs et les annotateurs se focalisent sur les mêmes types de termes : des composés néoclassiques, des termes anatomiques, des dérivés avec un sens spécialisé, et des valeurs numériques. Nous montrons en revanche que l'annotation manuelle présente plus de subjectivité, car elle repose sur une décision consciente de la part de l'annotateur par rapport aux résultats d'oculométrie, qui eux dépendent des processus et habitudes de lecture appris lors de l'apprentissage scolaire et familial. Les méthodes et le matériel d'oculométrie ont également des limitations actuellement, que nous indiquons dans le travail.

Dans les travaux futurs, il serait intéressant d'étudier le lien entre la longueur d'un texte et sa lisibilité et compréhension. L'hypothèse serait que les textes longs, même s'ils sont simplifiés, peuvent présenter des difficultés de lecture et de compréhension. L'impact d'autres facteurs (comme les définitions, les contextes favorables, les images et illustrations) peut également être étudié. À cause des contraintes expérimentales, des extraits courts de textes sont utilisés et étudiés. Pour cette raison, il serait intéressant d'effectuer des tests supplémentaires pour augmenter la variété des types de textes et de contenus traités. De plus, comme avec la simplification manuelle, l'efficacité de la simplification automatique peut aussi être testée et évaluée en utilisant des protocoles d'oculométrie.

Remerciements

Ce travail fait partie du projet *Termeye* financé par l'appel de l'Établissement de l'université Lille 3 et du projet *CLEAR (Communication, Literacy, Education, Accessibility, Readability)* financé par l'ANR sous la référence ANR-17-CE19-0016-01. Les extraits de textes médicaux proviennent du projet *RAVEL* financé par l'ANR sous la référence ANR-11-TECS-012. Nous remercions les participants de l'étude qui ont bien voulu nous consacrer leur temps. Nous remercions également les relecteurs anonymes qui ont permis d'améliorer la qualité de la version finale de ce papier.

Références

ALTARRIBA J., KROLL J., SHOLL A. & RAYNER K. (1996). The influence of lexical and conceptual constraints on reading mixed-language sentences : Evidence from eye fixations and naming times. *Memory and Cognition*, **24**, 477–92.

ANDREWS T. & COPPOLA D. (1999). Idiosyncratic characteristics of saccadic eye movements when viewing different visual environments. *Vision Res*, **39**(17), 2947–53.

ANTOINE E. & GRABAR N. (2016). Exploitation de reformulations pour l'acquisition d'un vocabulaire expert/non expert. In *Traitement Automatique des Langues Naturelles (TALN)*.

ASAAD W. & ANDEK MILLER G. R. (2000). Task-specific neural activity in the primate prefrontal cortex. *Neurophysiology*, **84**, 451–459.

BALAKRISHNA S. (2015). *Analyzing Text Complexity and Text Simplification : Connecting Linguistics, Processing and Educational Applications*. Thèse de doctorat, Eberhard Karls Universität Tübingen, Tübingen, Germany.

BERLAND G., ELLIOTT M., MORALES L., ALGAZY J., KRAVITZ R., BRODER M., KANOUSE D., MUNOZ J., PUYOL J. & ET AL M. L. (2001). Health information on the internet. accessibility, quality, and readability in english ans spanish. *JAMA*, **285**(20), 2612–2621.

BERNÈCHE F. & PERRON B. (2006). *Développer nos compétences en littératie : un défi porteur d'avenir. Enquête internationale sur l'alphabétisation et les compétences des adultes*. Rapport interne, Institut de la statistique du Québec, Canada.

BERTRAM R., KUPERMAN V., BAAYEN H. R. & HYÖNÄ J. (2011). The hyphen as a segmentation cue in triconstituent compound processing : It's getting better all the time. *Scandinavian Journal of Psychology*, **52**(6), 530–544.

BISSON M., VAN HEUVEN W., CONKLIN K. & TUNNEY R. (2014). Processing of native and foreign language subtitles in films : An eye-tracking study. *Applied Psycholinguistics*, **35**, 399–418.

BOZIC M., MARSLEN-WILSON W. D., STAMATAKIS E. A., DAVIS M. H. & TYLER L. K. (2007). Differentiating morphology, form, and meaning : Neural correlates of morphological complexity. *Journal of Cognitive Neuroscience*, **19**(9), 1464–1475.

CAIN K., TOWSE A. S. & KNIGHT R. S. (2009). The development of idiom comprehension : An investigation of semantic and contextual processing skills. *Journal of Experimental Child Psychology*, **102**(3), 280–298.

CHMIELIK J. & GRABAR N. (2011). Détection de la spécialisation scientifique et technique des documents biomédicaux grâce aux informations morphologiques. *TAL*, **51**(2), 151–179.

CLIFTON C. & STAUB A. (2011). Syntactic influences on eye movements in reading. In S. LIVERSEDGE, I. GILCHRIST & S. EVERLING, Eds., *The Oxford handbook of eye movements*, p. 895–909. Oxford University Press.

CLIFTON C., STAUB A. & RAYNER K. (2007). Eye movements in reading words and sentences. In R. VAN GOMPEL, M. FISCHER, W. MURRAY & R. HILL, Eds., *Eye Movements : A Window on Mind and Brain*. Oxford : Elsevier.

COOPER R. (1974). The control of eye fixation by the meaning of spoken language : A new methodology for the real-time investigation of speech perception, memory, and language processing. *Cognitive Psychol*, **6**, 84–107.

DOHMES P., ZWITSERLOOD P. & BÖLTE J. (2004). The impact of semantic transparency of morphologically complex words on picture naming. *Brain and Language*, **90**(1-3), 203–212.

DUCHOWSKI A. (2007). *Eye Tracking Methodology. Theory and practice*. London, UK : Springer.

ELTERMAN R., ABEL L., DAROFF R., DELL'OSSO L. & BORNSTEIN J. (1980). Eye movement patterns in dyslexic children. *J Learn Disabil*, **13**, 16–21.

FOX S. (2011). *Health topics. 80% of internet users look for health information online*. Rapport interne, Pew Internet & American Life Project, Washington DC.

FRANTZI K. T., ANANIADOU S. & MIMA H. (2000). Automatic recognition of multi-word terms : the C-Value/NC-Value method. *Int J on Digital Libraries*, **3**(2), 115–130.

FRENCK-MESTRE C. & PYNTE J. (1997). Syntactic ambiguity resolution while reading in a second and native languages. *The Quarterly Journal of Experimental Psychology*, **50**(A), 119–48.

GRABAR N. & HAMON T. (2016). Exploitation de la morphologie pour l'extraction automatique de paraphrases grand public des termes médicaux. *TAL*, **57**(1), 85–109.

HAMON T., ENGSTRÖM C. & SILVESTROV S. (2014). Term ranking adaptation to the domain : genetic algorithm based optimisation of the C-Value. In *PolTAL 2014*, p. 71–83.

HIGGINS E., LEINENGER M. & RAYNER K. (2014). Eye movements when viewing advertisements. *Front Psychol*, **5**, 210.

IACOBINI C. (2003). Composizione con elementi neoclassici. In M. GROSSMANN & F. RAINER, Eds., *La formazione delle parole in italiano*, p. 69–96. Walter de Gruyter.

JUCKS R. & BROMME R. (2007). Choice of words in doctor-patient communication : an analysis of health-related internet sites. *Health Commun*, **21**(3), 267–77.

KAGEURA K. & UMINO B. (1996). Methods of automatic term recognition. In *National Center for Science Information Systems*, p. 1–22.

KEATING G. (2009). Sensitivity to violations of gender agreement in native and non-native spanish : An eye-movement investigation. *Language Learning*, **59**, 503–35.

LÜDELING A., SCHMIDT T. & KIOKPASOGLOU S. (2002). Neoclassical word formation in german. *Yearbook of Morphology*, p. 253–283.

LÜTTMANN H., ZWITSERLOOD P. & BÖLTE J. (2011). Sharing morphemes without sharing meaning : Production and comprehension of german verbs in the context of morphological relatives. *Canadian Journal of Experimental Psychology/Revue canadienne de psychologie expérimentale*, **65**(3), 173–191.

MIELLET S., SPARROW L. & SERENO S. (2008). The effects of frequency and predictability in french : An evaluation of the e-z reader model. *Psychonomic Bulletin & Review*, **14**, 762–769.

MOLNAR F. (1981). About the role of visual exploration in aesthetics. In H. DAY, Ed., *Advances in Intrinsic Motivation and Aesthetics*, p. 385–414. New York : Plenum Press.

NILSSON BENFATTO M., ÖQVIST SEIMYR G., YGGE J., PANSELL T., RYDBERG A. & JACOBSON C. (2016). Screening for dyslexia using eye tracking during reading. *PLoS ONE*, **11**(12), e0165508.

PATEL V., BRANCH T. & AROCHA J. (2002). Errors in interpreting quantities as procedures : The case of pharmaceutical labels. *Int Journ Med Inform*, **65**(3), 193–211.

PLETNEVA N., VARGAS A. & BOYER C. (2011). *How do general public search online health information ?* Rapport interne, Health On the Net Foundation.

RAYNER K. (1998). Eye movements in reading and information processing : 20 years of research. *Psychological bulletin*, **124**(3), 372–373.

RAYNER K. & LIVERSEDGE S. (2004). *Visual and linguistic processing during eye fixations in reading*, In J. HENDERSON & F. FERREIRA, Eds., *The Interface of Language, Vision, and Action : Eye movements and the visual world*.

RELLO L., BAEZA-YATES R., BOTT S. & SAGGION H. (2013). Simplify or help ? : text simplification strategies for people with dyslexia. In ACM, Ed., *Int Cross-Disciplinary Conference on Web Accessibility*, p. 15–25.

RUBINO C. & MINDEN H. (1973). Analysis of eye-movements in children with reading disability. *Cortex*, **9**, 217–220.

SHARDLOW M. (2013). A comparison of techniques to automatically identify complex words. In *ACL Student Research Workshop*, p. 103–109.

SHARMIN S., SPAKOV O., RÄIHÄ K. & JAKOBSEN A. (2008). Effects of time pressure and text complexity on translators' fixations. In *ETNA*, p. 123–126.

SINGH A., MEHTA P., HUSAIN S. & RAJKUMAR R. (2016). Quantifying sentence complexity based on eye-tracking measures. In *Workshop on Computational Linguistics for Linguistic Complexity*, p. 202–212.

SPARROW L., MIELLET S. & COELLO Y. (2003). The effects of frequency and predictability on eye fixations in reading : An evaluation of the E-Z reader model. *Behavioral and Brain Sciences*, **26**, 503–505.

TANENHAUS M., SPIVEY-KNOWITON M., EBERHARDA K. & SEDIVY J. (1995). Integration of visual and linguistic information in spoken language comprehension. *Science*, **268**, 1632–1634.

TRAN T., CHEKROUD H., THIERY P. & JULIENNE A. (2009). Internet et soins : un tiers invisible dans la relation médecine/patient ? *Ethica Clinica*, **53**, 34–43.

TRUESWELL J., TANENHAUS M. & GARNSEY S. (1994). Semantic influences on parsing : Use of thematic role information in syntactic ambiguity resolution. *Journal of Memory and Language*, **33**, 285–318.

WENDT D., BRAND T. & KOLLMEIER B. (2014). An eye-tracking paradigm for analyzing the processing time of sentences with different linguistic complexities. *PLoS ONE*, **9**(6), e100186.

WILLIAMS M., PARKER R., BAKER D., PARIKH N., PITKIN K., COATES W. & NURSS J. (1995). Inadequate functional health literacy among patients at two public hospitals. *JAMA*, **274**(21), 1677–1682.

YANEVA V., TEMNIKOVA I. & MITKOV R. (2015). Accessible texts for autism : An eye-tracking study. In ACM, Ed., *Int ACM SIGACCESS Conference on Computers & Accessibility*, p. 49–57.

ZENG Q. T., TSE T., CROWELL J., DIVITA G., ROTH L. & BROWNE A. C. (2005). Identifying consumer-friendly display (CFD) names for health concepts. In *Ann Symp Am Med Inform Assoc (AMIA)*, p. 859–63.

Alignement de termes de longueurs variables en corpus comparables spécialisés

Jingshu Liu[1, 2] Emmanuel Morin[1] Sebastián Peña Saldarriaga[2]
(1) LS2N / Université de Nantes, 2 Chemin de la Houssinière, 44300 Nantes, France
(2) Dictanova, 6 rue René Viviani, 44200 Nantes, France
prénom.nom@ls2n.fr, prénom@dictanova.com

RÉSUMÉ

Nous proposons dans cet article une adaptation de l'approche compositionnelle étendue capable d'aligner des termes de longueurs variables à partir de corpus comparables, en modifiant la représentation des termes complexes. Nous proposons également de nouveaux modes de pondération pour l'approche standard qui améliorent les résultats des approches état de l'art pour les termes simples et complexes en domaine de spécialité.

ABSTRACT

Alignment of variable length terms in specialized comparable corpora

We propose in this paper an adaptation of the extended compositional approach able to align terms of variable lengths from comparable corpora, by modifying the representation of complex terms. We also propose new weighting modes for the standard approach that improve the results of state-of-the-art approaches for simple and complex terms in specialised domains.

MOTS-CLÉS : Multilinguisme, alignement, corpus comparables, vecteur de contexte.

KEYWORDS: Multilinguism, alignment, comparable corpora, context vector.

1 Introduction

L'extraction de lexiques bilingues à partir de corpus comparables a suscité de nombreux travaux depuis le début des années 90 (Fung, 1995; Rapp, 1999; Li & Gaussier, 2010; Morin & Daille, 2012; Mikolov *et al.*, 2013a; Xing *et al.*, 2015; Artetxe *et al.*, 2016; Hazem & Morin, 2016). Deux classes d'approches ont été développées selon la nature du terme à aligner. La première classe s'intéresse à l'alignement de mots et termes simples et repose sur des approches distributionnelles tandis que la seconde classe porte sur l'alignement de termes complexes et repose sur des approches compositionnelles. Peu de travaux se sont intéressés à proposer un cadre unifié permettant de réaliser l'alignement des termes simples et complexes en dehors de Delpech *et al.* (2012) pour l'alignement de termes simples vers des termes complexes. Notre objectif est de proposer un tel cadre unifié permettant l'alignement de termes de longueurs variables en domaine de spécialité.

Nous proposons d'adapter l'approche compositionnelle étendue pour prendre en compte les termes simples et complexes. En outre, nous proposons d'améliorer l'approche standard pour l'alignement de termes simples qui est exploitée dans l'approche compositionnelle étendue.

2 Approches standard et compositionnelle

Nous présentons dans cette section les approches existantes pour l'alignement de termes simples et complexes ainsi que les modifications apportées.

2.1 Approche standard

L'approche par alignement de contextes, appelée également approche standard (AS), est employée pour l'extraction de lexiques bilingues à partir de corpus comparables. Celle-ci repose sur la simple observation qu'un mot et sa traduction ont tendance à apparaître dans les mêmes contextes lexicaux (Fung, 1995; Rapp, 1999). Dans cette approche, il faut commencer par construire une matrice de cooccurrences pour les langues source et cible, où chaque ligne représente un vecteur de contexte dans une fenêtre de *n* mots. Ces vecteurs sont ensuite normalisés par exemple avec l'Information Mutuelle (IM). Il s'agit ensuite de transférer en langue cible le vecteur de contexte d'un mot en traduisant les éléments du vecteur via un dictionnaire bilingue. En ce qui concerne les mots qui ont plusieurs traductions, un poids est distribué en fonction de la fréquence de chaque traduction dans le corpus. Finalement les traductions candidates sont ordonnées en calculant la similarité du vecteur de contexte traduit avec l'ensemble des vecteurs de contexte en langue cible via une mesure de similarité comme le cosinus.

Avec l'AS, nous avons constaté que certains mots dans la fenêtre sont peu liés au terme central, en général, plus ce dernier est éloigné d'un mot du contexte, moins ils sont sémantiquement liés. Après le filtrage des mots outils, un mot à l'origine très éloigné du mot central peut apparaître dans la fenêtre. Cela rend le vecteur de contexte moins pertinent en tant que représentation. Afin de réduire cet effet, il nous faut une fonction de pondération qui satisfait quelques critères :

— La fonction doit être monotone décroissante en $[1, +\infty[$ étant donné que la distance ne peut jamais être inférieure à 1.

— L'image dans $[1, +\infty[$ doit représenter un poids dans $]0, 1]$.

— La fonction ne doit pas pénaliser la cooccurrence lorsque c'est déjà le plus proche du mot central. Autrement dit, la fonction renvoie 1 comme poids si la distance est égale à 1.

— L'écart entre les poids de pénalisation pour les distances longues doit être relativement petit car les mots éloignés du mot central ont une influence comparable en terme de contribution sémantique.

Il existe certainement plusieurs fonctions qui satisfont ces critères, dans ces travaux nous avons employé la fonction de poids g définie ainsi :

$$g(c|w) = \Delta(w, c)^{-\lambda}, \quad \lambda \in [0, 1] \tag{1}$$

où $g(c|w)$ est le poids du mot c dans le contexte de w, Δ la distance entre c et w et λ l'hyperparamètre qui détermine le degré de pénalisation (plus il est élevé, plus les contextes éloignés sont pénalisés). Notons que $\lambda = 0$ correspond à une distribution uniforme. La figure 1 montre le graphe de cette fonction quand λ est fixé à 0,25.

Munis de cette fonction de poids nous pouvons proposer une pondération en fonction de la distance (notée PFD) pour les mots dans la fenêtre :

$$PFD(w, c) = g(c|w) \times cooc(w, c) \tag{2}$$

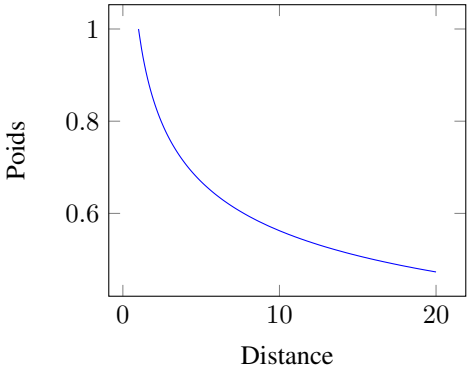


FIGURE 1: Fonction g avec $\lambda = \frac{1}{4}$

De nombreux travaux ont montré que l'IM surestime les faibles occurrences et sous-estime les hautes occurrences. Les travaux de Pennington *et al.* (2014) ont introduit une fonction pour lisser les occurrences sur laquelle nous nous appuyons pour proposer la mesure d'IM pondérée (IMP) pour améliorer l'IM dans nos expériences :

$$IMP(w,c) = f(cooc(w,c)) \times IM(w,c) \tag{3}$$

$$f(x) = \begin{cases} (x/x_{max})^{\alpha}, \alpha \in [0,1], \text{ si } x < x_{max} \\ 1 \text{ sinon} \end{cases} \tag{4}$$

où f est la fonction de Pennington *et al.* (2014), α et x_{max} sont les hyperparamètres qui caractérisent la fonction f. α détermine le degré de réduction pour les faibles occurrences, x_{max} détermine le seuil de réduction, par exemple $x_{max} = 20$ signifie que les occurrences moins de 20 doivent être réduites.

En raison de la petite taille des corpus en domaine spécialisé, les occurrences des mots ou des paires de mots ne sont pas toujours statistiquement fiables. Hazem & Morin (2016) ont montré que l'utilisation d'un corpus de langue générale peut améliorer significativement les résultats de l'AS. Ils proposent deux méthodes d'adaptation pour exploiter des ressources externes. La première adaptation, appelée approche standard globale (ASG), consiste à construire les vecteurs de contexte à partir d'un corpus comparable comportant des documents d'un domaine spécialisé et des documents du domaine général.

Nous avons implémenté la deuxième adaptation appelée approche standard sélective (ASS) qui a donné les meilleurs résultats. Pour chaque mot qui appartient au corpus du domaine spécialisé, s'il apparaît dans le corpus du domaine général, son vecteur de contexte spécialisés et généraux sont fusionnés. Cela permet de filtrer les mots de domaine général qui ne font pas partie du corpus spécialisé et rend l'approche standard sélective beaucoup moins coûteuse en temps calcul que l'approche standard globale. Soient S le vocabulaire du corpus spécialisé, G celui du corpus général, w est un mot à représenter, c est un mot qui apparaît dans la fenêtre autour de w :

$$\forall w \in S \cap G, \forall c \in S \cap G,\ cooc(w,c) = cooc_S(w,c) + cooc_G(w,c) \tag{5}$$

2.2 Approche compositionnelle

L'approche compositionnelle (AC) (Grefenstette, 1999; Tanaka, 2002; Robitaille *et al.*, 2006) est une approche simple et directe qui consiste à traduire chaque élément d'un terme complexe via un dictionnaire et à comparer toutes les permutations possibles par projection dans un corpus. La principale limite de cette approche est son incapacité à traduire un terme lorsqu'un des mots qui le composent n'est pas dans le dictionnaire. Pour résoudre ce problème, Morin & Daille (2012) ont proposé l'approche compositionnelle étendue (ACE), dont l'objectif est de combiner les avantages des approches standard et compositionnelle en substituant les mots hors dictionnaire par leur vecteur de contexte obtenu par l'AS. L'ACE commence par construire la matrice de cooccurrences comme pour l'approche standard. Ensuite il s'agit d'appliquer une traduction directe renforcée par alignement de contexte. Si un mot d'un terme à traduire n'est pas présent dans le dictionnaire, nous utilisons le vecteur de contexte obtenu par l'AS et le traduisons en langue cible, sinon nous récupérons directement le vecteur de contexte de la traduction en langue cible. L'étape suivante est la génération de toutes les combinaisons de la représentation pour un terme en langue source. Finalement les termes candidats sont ordonnés suivant le calcul de similarité avec tous les termes de même longueur en langue cible, le score final pour chaque possibilité étant défini par la moyenne arithmétique ou géométrique de chaque score de similarité.

2.3 Adaptation à l'alignement de termes de longueurs variables

L'approche compositionnelle étendue ne permet pas de prendre en compte le problème de fertilité, c'est-à-dire l'alignement de termes de longueurs variables. Par exemple, le terme anglais « *wind vane* » peut être traduit par « *girouette* » en français et le terme anglais « *wind energy* » par « *Windenergie* » en allemand. Afin de prendre en compte ces cas, nous nous sommes inspirés des travaux de Blacoe & Lapata (2012) dans lesquels la représentation d'une phrase est la somme des représentations distributionnelles de chacun de ses mots. Cependant dans ce travail la pondération de chaque élément d'un terme complexe n'est pas prise en compte. En effet le vecteur construit par la simple somme de chaque élément est davantage orienté vers les vecteurs qui ont des valeurs plus importantes. Dans ce travail nous souhaitons vérifier l'hypothèse que la moyenne des vecteurs normalisés de tous les éléments représente plus fidèlement un terme complexe. L'intuition derrière cette hypothèse est que ces éléments de longueur uniforme assurent un impact équivalent pour la construction sémantique d'un terme complexe. Nous proposons ainsi de modifier la représentation des termes complexes dans l'ACE ainsi que dans l'AS :

$$vecteur(terme) = \frac{1}{n}\sum_{i}^{n}\frac{vecteur(w_i)}{||vecteur(w_i)||} \tag{6}$$

où $||\vec{x}||$ représente la l^2-$norme$ d'un vecteur $\vec{x}$ et n la longueur du terme.

Dans la figure 2 nous démontrons la logique de la moyenne sur les vecteurs normalisés par un exemple : soit un terme complexe composé de deux mots a et b, et leurs vecteurs de contexte respectifs $\vec{a}$ et $\vec{b}$. Si $\vec{a}$ est plus long que $\vec{b}$ (dans le contexte de l'approche standard cela signifie que les cooccurrences concernant le mot a sont plus importantes), la moyenne des deux vecteurs de contexte sera plus proche de a. Or il n'est pas toujours vrai que le sens d'un terme complexe est déterminé par l'élément plus fréquent. Le vecteur final que nous proposons, illustré en rouge sur la figure 2, forme un angle égal avec $\vec{a}$ et $\vec{b}$. Dans la partie basse du graphe nous illustrons la différence (l'angle) entre les deux

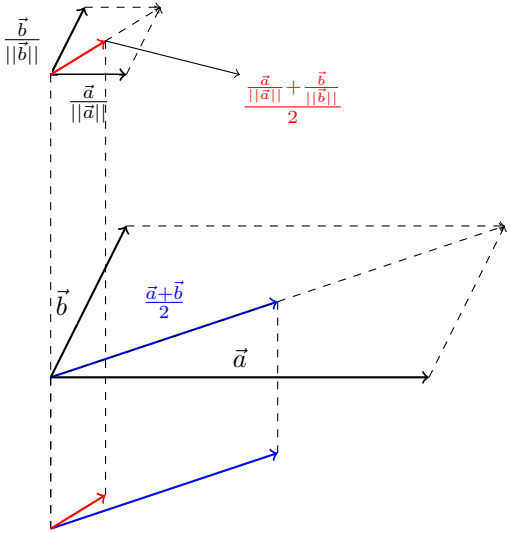


FIGURE 2: Illustration de la différence entre la moyenne des vecteurs non normalisés (le vecteur en bleu) et normalisés (le vecteur en rouge).

vecteurs obtenus par les deux stratégies. Il est par ailleurs clair que si nous calculons la similarité par le Cosinus, la moyenne et la somme des vecteurs sont équivalentes car le Cosinus est une mesure en fonction de la direction seule.

La représentation du terme se fait dans un seul vecteur, donnant la capacité de gérer les traductions de longueurs variables tout en réduisant le temps de calcul. En effet, dans l'ACE originale, aligner un terme complexe demande de calculer toutes les permutations possibles, il faut donc comparer un nombre factoriel de vecteurs, contre un seul vecteur à comparer pour la version adaptée.

2.4 Adaptation de l'approche compositionnelle étendue

Nous illustrons le schéma de notre méthode qui consiste en l'approche compositionnelle avec l'adaptation de la représentation pour les termes complexes.

Dans la figure 3 nous montrons comment un terme complexe en langue source est aligné à un terme de longueur différente en langue cible. w_{sn} est le n-ième mot du terme complexe en langue source, w_{sn} le n-ième mot du terme en langue cible. Dans l'exemple de la figure 3, 3 traductions ont été trouvées pour w_{s1} dans le corpus comparable via le dictionnaire, une pour w_{s2} et aucune traduction n'a été trouvé pour w_{s3} dans le corpus comparable. t_n^m signifie le vecteur de contexte pour m-ième traduction possible en langue cible pour le mot w_{sn}, alors que s^n représente le vecteur de contexte obtenu par l'approche standard pour le mot w_{sn}.

Il est à noter que tous ces vecteurs sont dans l'espace commun de la langue cible. Les opérations par élément telle que l'addition sont donc raisonnables. Lors de l'étape suivante 3 ($\prod m$) traductions sont possibles pour le terme complexe parce que nous ne prenons pas en compte l'ordre des éléments du terme complexe. Nous appliquons notre méthode de la représentation des termes complexes et nous obtenons un seul vecteur pour chaque traduction.

Ensuite pour comparer la similarité entre une possible traduction du terme en langue source $w_{s1}w_{s2}w_{s3}$ avec le terme en langue cible $w_{t1}w_{t2}$, il suffit de calculer le produit scalaire (nous supposons que les vecteurs sont déjà normalisés après le calcul de la moyenne). Finalement la similarité entre les deux termes est $max(similarité1, similarité2, similarité3)$.

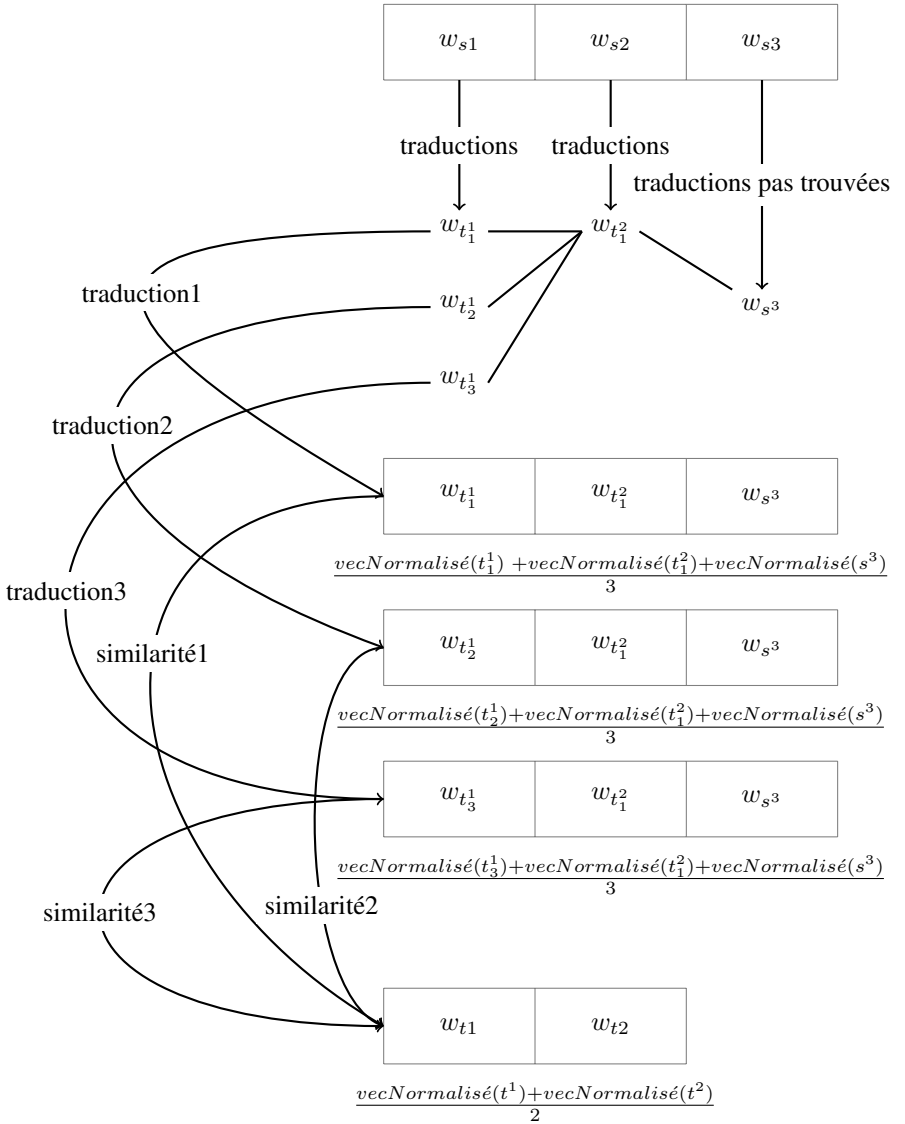


FIGURE 3: Schéma de notre approche pour l'alignement unifié

3 Expériences

Afin de valider notre implémentation des méthodes de l'état de l'art, nous les testons sur les termes simples (TS) avant de les appliquer au sujet qui nous intéresse : les termes complexes (TC). Nous avons aussi expérimenté l'utilisation de l'ASS dans l'approche compositionnelle étendue. À notre connaissance, notre proposition est le premier travail à les combiner.

3.1 Ressources

Nous avons utilisé le corpus spécialisé *Breast Cancer* (BC) (Hazem & Morin, 2016) pour l'expérience sur les termes simples et le corpus de langue générale *News Commentary* (NC) [1]. En ce qui concerne l'alignement des termes complexes, nous avons utilisé le corpus spécialisé *Wind Energy* (WE) [2] et deux corpus propriétaires spécialisés traitant des domaines du luxe et de la cosmétique.

Afin d'évaluer l'alignement des termes simples, nous avons utilisé la liste de référence fournie par Hazem & Morin (2016) pour le corpus BC. En ce qui concerne les termes complexes, nous avons trois listes de référence, celle du corpus WE est construite à partir des listes terminologiques fournies avec le corpus, et celles des corpus luxe et cosmétique sont construites manuellement par des experts du domaine à partir d'une liste de termes simples et complexes extraits par un système propriétaire symbolique qui utilise des règles morpho-syntaxiques comme *ACABIT* (Daille, 2003) [3] ou *TermSuite* (Daille, 2016) [4]. Les traductions sont validées en 3 itérations par les même experts du domaine qui ont choisi les termes à traduire. De multiples traductions pour un terme source sont possibles dans la liste de référence pour les corpus WE, Luxe et Cosmétique. Nous avons inclus les termes simples car leurs traductions peuvent être des termes complexes, ils s'intègrent donc naturellement dans l'alignement des termes complexes. De plus les termes à traduire dans les corpus pour la tâche des termes complexes sont hors dictionnaire.

Les candidats pour nos expériences sur les termes simples sont tous les mots dans le vocabulaire, pour nos expériences sur les termes complexes les candidats sont tous les mots dans le vocabulaire plus tous les termes complexes extraits par le système d'extraction terminologique qui génère généralement trois fois plus de termes complexes que de mots dans le vocabulaire.

Le tableau 1 présente les principales caractéristiques de ces différentes ressources.

Corpus	Nombre de mots		Taille de vocabulaire		Référence
	FR	EN	FR	EN	
BC	521 262	525 934	6 630	8 821	TS : 248
WE	314 549	313 943	6 038	7 134	TC : 73
Luxe	101 542	139 867	3 064	3 981	TC : 276, TS : 13
Cosmétique	430 106	837 579	3 913	5 592	TC : 185, TS : 14
NC	5,7 M	4,7 M	23 597	29 489	

Tableau 1: Caractéristiques des corpus utilisés

1. opus.lingfil.uu.se
2. ttc.project.eu
3. www.bdaille.com/index.php?option=com_content&task=blogcategory&id=5&Itemid=5
4. termsuite.github.io

3.2 Configuration

Pour toutes nos expériences, nous prétaitons les corpus via la tokenisation, le pos-tagging et la lemmatisation, ensuite nous filtrons les hapax et les mots outils. Dans la PFD, l'hyperparamètre a été empiriquement fixé à $\lambda = 0,25$. La taille de la fenêtre contextuelle d'un terme est de trois mots avant et trois mots après. Dans l'IMP, l'hyperparamètres x_{max} est fixé à 20 car la taille de nos corpus est relativement petite. L'autre hyperparamètre α de l'IMP est fixé à $\frac{3}{4}$ comme Pennington *et al.* (2014) ont décidé dans leur travaux. Il est d'ailleurs intéressant que dans les travaux de Mikolov *et al.* (2013b), une même puissance fractionnelle a été introduite pour obtenir les meilleurs résultats.

Pour la traduction des vecteurs de contexte nous avons utilisé le dictionnaire FRAN-EURADIC de ELRA [5] comportant 243 539 entrées.

Afin d'accélérer le processus pendant la phase de comparaison, tous nos vecteurs sont normés à 1 et nous avons utilisé la similarité cosinus car elle permet d'utiliser la parallélisation sur CPU ou GPU.

3.3 Résultats

Le tableau 2 montre les résultats de l'AS et l'ASS pour les termes simples. Nous voyons que l'IM pondérée améliore les résultats par rapport à ceux obtenus par Hazem & Morin (2016) avec l'IM mais aussi à ceux qu'ils obtiennent avec le *DOR* (*Discounted Odds Ratio*) (Evert, 2005) à la place du cosinus, avec le DOR la MAP atteint 0,270. Nos résultats montrent l'intérêt de pénaliser les petites occurrences pour compenser la surestimation de l'IM originale. Cependant cette pondération est moins efficace quand les données sont enrichies car la surestimation des petites occurrences est lissée par l'ajout de données exogènes. L'IM pondérée avec les données exogènes déchoit quelques bonnes traductions de la position top 1 (P@1 de 50,8 à 48,0), mais elle promeut plus de bonnes traductions dans le top 5 (P@5 de 62,5 à 64,1). Étant donné que certains termes à traduire sont assez peu fréquents ou même inexistants dans le corpus général, il est possible que pénaliser toutes les petites occurrences réduise des traits discriminants du corpus général. Pourtant nous constatons que la PFD améliore systématiquement nos résultats dans tous les cas. Si nous regardons la comparaison entre l'AS et l'AS + PFD, et celle entre l'ASS et l'ASS + PFD, les résultats sont meilleurs pour l'ensemble des quatre mesures.

Modèle	P@1	P@5	P@20	MAP
AS (Hazem & Morin, 2016)	18,5	35,5	46,0	25,9
AS + IMP	21,4	**37,1**	49,6	28,9
AS + PFD	19,4	36,3	47,2	27,4
AS + IMP + PFD	**22,6**	**37,1**	**50,4**	**29,5**
ASS	50,8	62,5	72,6	56,5
ASS + IMP	48,0	**64,1**	71,8	55,0
ASS + PFD	**51,6**	62,9	**73,4**	**57,3**
ASS + IMP + PFD	48,4	**64,1**	**73,4**	55,8

Tableau 2: Précision@k et MAP (%) pour l'alignement des termes simples sur le corpus BC.

En ce qui concerne les termes complexes, nous effectuons trois ensembles d'expériences reposant sur

5. www.elra.info

l'approche standard (AS), l'approche standard sélective (ASS), l'approche compositionnelle (AC).

Pour vérifier si l'IM pondérée est toujours favorable dans différents scénarios, nous l'appliquons à plusieurs configurations.

Afin d'illustrer la capacité de notre algorithme à aligner des termes de longueurs variables, le tableau 3 montre quelques exemples de traductions ou quasi-traductions trouvées par notre méthode avec l'adaptation de la représentation pour les termes complexes. Ces traductions sont dans le top@5 des candidats pour les différents corpus. Parmi toutes les traductions dans la liste top 5, nous montrons uniquement celles dont la longueur est différente du terme de départ.

Corpus	Alignements (anglais → français)
Luxe	sneaker shoe → sneaker invoice → facture d'achat
Cosmétique	wide variety of choice → nombreux choix very small tight store → petit magasin
WE	greenhouse gas → gaz à effet de serre power system → système éolien de puissance

Tableau 3: Exemples des traductions trouvées dans le top@5

Le tableau 4 montre les résultats des différentes méthodes avec l'adaptation pour les termes complexes. Nous voyons qu'elles se comportent de façon homogène à travers les différents domaines, tant que l'approche utilise des vecteurs de contexte unifiés. Nous ne montrons pas les résultats de l'AS sans l'adaptation de la représentation de TC car ceux-ci sont négligeables. En ce qui concerne le temps de calcul, l'alignement d'un terme complexe dans le corpus Luxe prend en moyenne 10 minutes avec l'ACE alors que la version adaptée prend 20 secondes. De plus la version originale ne permet pas d'aligner les termes de longueurs variables, alors que la traduction pour les termes complexes de la même longueur n'existe pas toujours dans le corpus comparable. Nous avons décidé de ne pas reproduire les résultats de l'ACE originale sur les autres corpus étant donné les performances supérieures de la version adaptée et le temps de calcul beaucoup plus rapide.

Dans le tableau 4 nous observons que nos propositions, l'IMP et la PFD, améliorent significativement (de 3 à 10 points en MAP) les résultats par rapport à l'approche standard avec ou sans données exogènes. L'ASS ne fonctionne pas aussi bien que pour les termes simples. En général, l'approche compositionnelle étendue avec l'adaptation pour les TC présente les meilleurs résultats à travers les trois corpus de différents domaines de spécialité. Sur le corpus Luxe nous avons de plus expérimenté l'approche compositionnelle étendue sans l'adaptation : son amélioration sur l'approche compositionnelle est de 4,2 points en MAP, ce qui est relativement faible par rapport à la version avec l'adaptation qui est de 15,4 points.

3.4 Discussion

Contrairement aux résultats sur les termes simples, l'IM pondérée améliore toujours les résultats avec l'ASS. Nous expliquons cela par le fait que le vecteur pour un terme complexe est une moyenne, par conséquent le risque que chaque élément soit peu fréquent est beaucoup moins élevé. La pénalisation pour les faibles occurrences devient donc une pénalisation partielle.

Corpus	Modèle	P@1	P@5	P@20	MAP
Luxe	AS	4,2	11,8	24,2	8,9
	AS + IMP	14,2	21,4	33,2	18,6
	AS + IMP + PFD	**14,5**	**21,5**	**33,9**	**18,9**
	AC	23,8	25,6	25,6	24,7
	ACE[a] + IMP + PFD	24,6	32,2	39,1	28,9
	ACE + IMP + PFD	**34,6**	**44,6**	**57,1**	**40,1**
	ACE[b] + IMP + PFD	24,2	37,0	51,2	31,2
	ASS	1,7	3,1	8,7	3,0
	ASS + IMP	2,8	8,0	13,8	5,8
	ASS + IMP + PFD	**3,1**	**8,3**	**14,5**	**5,9**
Cosmétique	AS	0,5	4,0	8,0	2,9
	AS + IMP	**5,6**	11,1	21,6	**10,3**
	AS + IMP + PFD	4,5	**12,6**	**22,6**	**10,3**
	AC	**12,5**	**19,1**	19,6	15,4
	ACE + IMP + PFD	11,6	**19,1**	**28,1**	**16,8**
	ACE[b] + IMP + PFD	7,0	16,6	24,6	12,9
	ASS	0	1,0	5,6	1,0
	ASS + IMP	**3,5**	9,5	17,1	7,1
	ASS + IMP + PFD	3,0	**10,0**	**17,6**	**7,3**
WE	AS	1,4	24,7	43,8	12,2
	AS + IMP	**12,3**	28,8	**50,7**	21,7
	AS + IMP + PFD	**12,3**	**31,5**	**50,7**	**21,9**
	AC	**59,0**	68,5	68,5	61,5
	ACE + IMP + PFD	42,5	**80,8**	**89,0**	60,0
	ACE[b] + IMP + PFD	53,4	**87,7**	**90,4**	**66,3**
	ASS	**11,0**	20,5	31,5	14,8
	ASS + IMP	9,6	**27,4**	37,0	**17,9**
	ASS + IMP + PFD	8,2	**27,4**	**40,0**	17,8

Tableau 4: Précision@k et MAP (%) pour l'alignement des termes complexes ([a] sans adaptation pour les TC de longueurs variables et [b] avec ASS pour l'alignement des mots hors dictionnaire).

Il est d'ailleurs surprenant que l'enrichissement n'apporte pas d'amélioration substantielle pour les corpus Luxe et Cosmétique, qui sont très bruités et contiennent beaucoup de mots hors dictionnaire (des argots des internautes). Notre hypothèse est qu'avec l'introduction des ressources externes, les mots de langue générale submergent le sens des mots spécifiques au corpus original. Dans nos expériences, le style du corpus général NC diffère beaucoup des corpus extraits des utilisateurs de l'internet. En effet, les mots spécifiques sont absents du corpus NC ou ne se comportent pas de la même manière dans celui-ci. Sur les deux corpus propriétaires extraits de l'internet les meilleurs résultats sont obtenus avec l'approche ACE adaptée sans ASS. Une solution potentielle est d'utiliser davantage de corpus généraux contenant plus de mots spécifiques. Pour le corpus WE qui est un corpus plus propre, et dont la plupart des composants des termes se trouvent dans le dictionnaire, la partie AS apporte peu par rapport à l'approche compositionnelle, et le classement par fréquence de l'AC est meilleur en P@1 et en MAP. Dans ce même corpus l'ASS dans l'ACE améliore sensiblement la MAP par rapport aux approches sans données exogènes, cela corrobore notre intuition sur l'enrichissement des données qui permet d'améliorer l'AS pour les mots qui existent dans les deux corpus et possèdent

des distributions similaires.

Parmi les termes complexes pour lesquels nous n'avons pas de traductions, nous identifions trois catégories de causes d'erreur :

— Faible compositionnalité. La traduction de l'ensemble n'est pas une combinaison de toutes les traductions de chaque élément. Par exemple « *pitch angle* » est traduit par « *angle d'inclinaison* » par le système alors que la bonne traduction est « *angle de calage* » où le mot « *calage* » n'est pas la traduction d'un mot dans le dictionnaire bilingue. Ce problème peut être encore plus grave entre deux autres langues linguistiquement lointaines, Tanaka (2002) rapporte qu'au moins 50% des composés japonais *NN* ne sont pas traduits en anglais par le même modèle syntaxique de composition. C'est d'ailleurs une piste intéressante à poursuivre pour expérimenter si notre système est capable de maintenir la performance sur d'autres langues.

— Ambiguïté. Plusieurs traductions possibles ont la même similarité dans la liste top@k car chaque mot composant a plusieurs traductions dans le dictionnaire et certaines combinaisons existent dans la liste des candidats terminologiques. Par exemple « *collier dior* » est traduit par « *chain* » alors que la bonne traduction est « *dior necklace* » ou « *necklace* » si nous permettons les quasi-traductions. Les mots « *chain* » et « *necklace* » sont tous deux traductions dans le dictionnaire or dans le contexte spécifique, « *chain* » est une fausse traduction. C'est une des causes qui engendrent la randomisation de l'ordre des 5 premiers candidats pour certains termes car plusieurs traductions candidates ont le même score. Cela peut expliquer le fait que l'ACE présente un résultat P@1 pour le corpus Cosmétique et WE inférieur à l'AC. Une couche de plus de désambiguïsation selon le contexte pour les mots qui ont plusieurs traductions dans le corpus pourrait être une solution intéressante.

— Différents ordres avec les même mots. Notre approche ne prend pas compte l'ordre des mots dans les termes complexes. De ce fait plusieurs termes complexes ont le même vecteur de représentation. Par exemple « *power installation* » est traduit par « *puissance d'installation* » alors que la bonne traduction est « *installation de puissance* ». C'est la deuxième cause qui rend l'ordre des 5 premiers candidats aléatoire pour certains termes. C'est aussi pourquoi l'ACE n'est pas meilleure en top@1 par rapport à l'AC pour le corpus Cosmétique et WE, mais elle rattrape et même surpasse l'AC en top@5. Cependant prendre en compte l'ordre provoquerait une augmentation significative du temps de calcul et l'alignement de termes de longueurs variables serait alors plus difficilement réalisable.

4 Conclusion

Nous avons proposé dans cet article une adaptation de l'ACE capable d'aligner des termes de longueurs variables, en modifiant la représentation des termes complexes. Nous avons aussi proposé de nouveaux modes de pondération pour l'AS qui améliorent les résultats des approches état de l'art pour les termes simples et complexes en domaine de spécialité. Nous espérons que les contributions de cet article aideront à approfondir la compréhension des approches par vecteurs de contexte.

L'enrichissement avec des données externes n'a pas montré des résultats homogènes pour l'alignement de termes complexes, cela nous donne des pistes pour notre futur travail. D'un côté nous prévoyons de travailler sur la représentation unifiée, qui est aujourd'hui simpliste car elle considère que chaque composant d'un terme complexe a la même importance dans la constitution du sens global. De l'autre côté, nous envisageons de profiter des système neuronaux, bien que ces méthodes ne s'accordent pas

naturellement avec notre contexte spécifique, cela nous semble intéressant de réfléchir à de nouvelles architectures correspondant mieux à notre besoin.

Remerciements

Les auteurs tiennent à remercier M. Joseph Lark pour ses commentaires et propositions qui ont permis d'étoffer le texte, ainsi que les 3 relecteurs anonymes pour leurs remarques pertinentes.

Références

ARTETXE M., LABAKA G. & AGIRRE E. (2016). Learning principled bilingual mappings of word embeddings while preserving monolingual invariance. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing (EMNLP'16)*, p. 2289–2294, Austin, TX, USA.

BLACOE W. & LAPATA M. (2012). A comparison of vector-based representations for semantic composition. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP'12)*, p. 546–556, Jeju Island, Korea.

DAILLE B. (2003). Conceptual structuring through term variations. In *Proceedings ACL 2003 Workshop on Multiword Expressions : Analysis, Acquisition and Treatment*, p. 9–16, Sapporo, Japan.

DAILLE B. (2016). Terminology extraction with term variant detection. In *Proceedings of ACL-2016 System Demonstrations*, p. 13–18, Berlin, Germany.

DELPECH E., DAILLE B., MORIN E. & LEMAIRE C. (2012). Extraction of domain-specific bilingual lexicon from comparable corpora : compositional translation and ranking. In *Proceedings of the 24th International Conference on Computational Linguistics (COLING'12)*, p. 745–762, Mumbai, India.

EVERT S. (2005). *The statistics of word cooccurrences : word pairs and collocations*. PhD thesis, University of Stuttgart.

FUNG P. (1995). Compiling bilingual lexicon entries from a non-parallel english-chinese corpus. In *Proceedings of the 3rd Annual Workshop on Very Large Corpora (VLC'95)*, p. 173–183, Cambridge, MA, USA.

GREFENSTETTE G. (1999). The world wide web as a resource for example-based machine translation tasks. In *Proceedings of the ASLIB Conference on Translating and the Computer 21*, London, UK.

HAZEM A. & MORIN E. (2016). Efficient Data Selection for Bilingual Terminology Extraction from Comparable Corpora. In *Proceedings of the 26th International Conference on Computational Linguistics (COLING'16)*, p. 3401–3411, Osaka, Japan.

LI B. & GAUSSIER E. (2010). Improving corpus comparability for bilingual lexicon extraction from comparable corpora. In *Proceedings of the 23rd International Conference on Computational Linguistics (COLING'10)*, p. 644–652, Beijing, China.

MIKOLOV T., LE Q. V. & SUTSKEVER I. (2013a). Exploiting similarities among languages for machine translation. *CoRR*, **abs/1309.4168**.

MIKOLOV T., SUTSKEVER I., CHEN K., CORRADO G. & DEAN J. (2013b). Distributed representations of words and phrases and their compositionality. In *Proceedings of the 26th International Conference on Neural Information Processing Systems (NIPS'13)*.

MORIN E. & DAILLE B. (2012). Revising the Compositional Method for Terminology Acquisition from Comparable Corpora. In *Proceedings of the 24th International Conference on Computational Linguistics (COLING'12)*, p. 1797–1810, Mumbai, India.

PENNINGTON J., SOCHER R. & MANNING C. (2014). Glove : Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP'14)*, p. 1532–1543, Doha, Qatar.

RAPP R. (1999). Automatic Identification of Word Translations from Unrelated English and German Corpora. In *Proceedings of the 37th Annual Meeting of the Association for Computational Linguistics (ACL'99)*, p. 519–526, College Park, MD, USA.

ROBITAILLE X., SASAKI Y., TONOIKE M., SATO S. & UTSURO T. (2006). Compiling French-Japanese Terminologies from the Web. In *Proceedings of the 11th Conference of the European Chapter of the Association for Computational Linguistics (EACL'06)*, p. 225–232, Trento, Italy.

TANAKA T. (2002). Measuring the Similarity Between Compound Nouns in Different Languages Using Non-parallel Corpora. In *Proceedings of the 19th International Conference on Computational Linguistics (COLING'02)*, p. 1–7, Taipei, Taiwan.

XING C., WANG D., LIU C. & LIN Y. (2015). Normalized Word Embedding and Orthogonal Transform for Bilingual Word Translation. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics : Human Language Technologies (NAACL'15)*, p. 1006–1011, Denver, CO, USA.

Étude de la reproductibilité des *word embeddings* : repérage des zones stables et instables dans le lexique

Bénédicte Pierrejean Ludovic Tanguy
CLLE-ERSS : CNRS & Université de Toulouse
{benedicte.pierrejean, ludovic.tanguy}@univ-tlse2.fr

RÉSUMÉ
Les modèles vectoriels de sémantique distributionnelle (ou *word embeddings*), notamment ceux produits par les méthodes neuronales, posent des questions de reproductibilité et donnent des représentations différentes à chaque utilisation, même sans modifier leurs paramètres. Nous présentons ici un ensemble d'expérimentations permettant de mesurer cette instabilité, à la fois globalement et localement. Globalement, nous avons mesuré le taux de variation du voisinage des mots sur trois corpus différents, qui est estimé autour de 17% pour les 25 plus proches voisins d'un mot. Localement, nous avons identifié et caractérisé certaines zones de l'espace sémantique qui montrent une relative stabilité, ainsi que des cas de grande instabilité.

ABSTRACT
Reproducibility of word embeddings : identifying stable and unstable zones in the semantic space

Distributional semantic models trained using neural networks techniques yield different models even when using the same parameters. We describe a series of experiments where we examine the instability of word embeddings both from a global and local perspective for several models trained with the same parameters. We measured the global variation for models trained on three different corpora. This variation is estimated to about 17% for the 25 nearest neighbours of a target word. We also identified and described local zones of stability and instability in the semantic space.

MOTS-CLÉS : plongements lexicaux, évaluation, stabilité, reproductibilité.

KEYWORDS: word embeddings, evaluation, stability, reproducibility.

1 Introduction

La popularité des modèles prédictifs en sémantique distributionnelle est indéniable, et les outils comme *Word2vec* (Mikolov *et al.*, 2013) ont su s'imposer comme de nouveaux standards à la fois dans les applications du TAL et dans les investigations empiriques des différentes facettes de la sémantique lexicale. Ces outils permettent de représenter de façon compacte des unités lexicales (par des vecteurs de quelques centaines de dimensions), en se basant uniquement sur des quantités de texte correspondant aux standards contemporains (centaines de millions ou milliards de mots) et d'obtenir par ce biais des rapprochements convaincants d'unités lexicales sémantiquement liées.

Toutefois, au-delà des différentes méthodes mises en œuvre dans ces techniques, leur utilisation soulève un ensemble de questions à la fois méthodologiques et pratiques. Si les embeddings (ou

plongements lexicaux) ont prouvé leur intérêt, et sont utilisés sans hésitation pour représenter des contenus lexicaux en amont d'une application de TAL (notamment basée sur des méthodes d'apprentissage profond) (Goldberg, 2016), plusieurs questions restent en suspens quant aux conditions d'utilisation de ces techniques.

Depuis longtemps on sait que les méthodes d'analyse distributionnelle sont très sensibles à un ensemble de paramètres. On se base généralement sur des évaluations intrinsèques pour estimer la valeur optimale des paramètres, en comparant par exemple les similarités entre les mots au sein des modèles distributionnels avec celles estimées par des humains (Faruqui *et al.*, 2016), et ce sur un échantillon réduit (généralement quelques centaines de paires de mots) ce qui soulève des problèmes concernant la fiabilité des évaluations effectuées.

Les modèles distributionnels neuronaux de type *Word2vec* soulèvent d'autres questions liées à la *reproductibilité* des modèles construits, puisqu'ils se basent sur des méthodes stochastiques à différentes étapes de leur traitement. Autrement dit, pour un même outil paramétré de la même façon et appliqué à un même corpus, un modèle différent est généré à chaque exécution. Cette instabilité doit bien entendu être prise en compte, surtout lorsque l'on utilise les modèles distributionnels à des fins d'exploration locale d'une partie du lexique ou d'un corpus particulier (Hellrich & Hahn, 2016; Antoniak & Mimno, 2018). La table 1 donne quelques exemples des mots les plus similaires de deux mots-cibles pour trois de ces modèles construits de façon identique (voir en 3.1 pour les détails de ceux-ci). On y voit que si globalement les différents voisins sont sémantiquement pertinents, certains d'entre eux n'apparaissent en bonne place que dans certains modèles (e.g. *sheaf* et *newspaper* pour *paper*, *pink* et *red* pour *white*), et que les variations ne semblent pas du même ordre pour les deux mots-cibles choisis.

Mot-cible	paper (n)			white (adj)		
Rang	Modèle 1	Modèle 2	Modèle 3	Modèle 1	Modèle 2	Modèle 3
1	magazine	book	book	black	black	black
2	book	magazine	sheaf	grey	grey	yellow
3	pamphlet	newspaper	magazine	blue	yellow	grey
4	journal	pamphlet	folder	yellow	red	blue
5	article	folder	parchment	pink	blue	red

TABLE 1. Exemple de la variation des 5 plus proches voisins du nom *paper* et de l'adjectif *white* dans deux modèles distributionnels construits de façon identique

Nous proposons dans cet article une série d'études visant à éclaircir ce point sous différents angles, en traitant les questions suivantes :

— Comment peut-on quantifier la stabilité interne d'un modèle distributionnel (ou plus précisément d'une *configuration* de modèle distributionnel) en comparant plusieurs instances produites dans des conditions identiques ?
— La variation d'une instance d'un modèle à une autre est-elle répartie uniformément dans l'espace vectoriel, ou au contraire y a-t-il des zones de plus grande stabilité ? Si oui, peut-on identifier et caractériser ces zones ?
— Comment la stabilité interne d'une configuration varie-t-elle lorsque l'on change les paramètres, notamment lorsque l'on change le corpus sur lequel le modèle est construit ?

La Section 2 présente un bref aperçu des travaux similaires portant sur l'évaluation et sur la reproductibilité des *embeddings*. Nous présentons en Section 3 notre dispositif expérimental et les choix faits

pour mesurer la variation entre deux modèles. Les deux dernières sections (4 et 5) sont consacrées à l'exploration globale puis locale de cette variation, notamment en identifiant les zones du lexique qui sont le plus (et le moins) sujettes à variation.

2 Travaux similaires

2.1 Evaluation des embeddings

L'étude de l'impact des différents paramètres sur la qualité relative des modèles distributionnels a donné lieu à de nombreuses publications. Il a ainsi été montré que le changement de paramètres peut entraîner des modifications importantes. Ces paramètres peuvent correspondre aux familles de méthodes de construction de modèles distributionnels (Bernier-Colborne & Drouin, 2016; Chiu *et al.*, 2016), aux hyperparamètres de ces méthodes, aux corpus d'apprentissage (nature, taille, prétraitements) (Asr *et al.*, 2016; Sahlgren & Lenci, 2016) ou encore aux types de contextes utilisés pour représenter les unités lexicales (Levy & Goldberg, 2014; Melamud *et al.*, 2016; Li *et al.*, 2017). Ces travaux abordent généralement la comparaison par le biais de bancs de test d'évaluation intrinsèque d'un modèle en comparant les similarités qui y sont calculées avec des mesures relevant de jugements humains. Bien que pratiques et peu coûteuses, ces méthodes d'évaluation présentent plusieurs inconvénients. Les jeux d'évaluation utilisés posent notamment des problèmes de subjectivité (scores plus élevés pour des mots qui sont associés comparé à des mots similaires) ou encore de surapprentissage (même jeux d'évaluation utilisés jusqu'à obtenir les résultats attendus) (Faruqui *et al.*, 2016). De plus, les jeux d'évaluation ne permettent d'évaluer qu'une partie d'un modèle puisqu'ils sont généralement constitués de quelques centaines de paires de mots.

Plusieurs alternatives ont été envisagées pour pallier à ces limites. La première est de faire appel à des évaluations extrinsèques en aval. Ces méthodes consistent à intégrer les embeddings dans un système de TAL dont la performance globale sera évaluée (classification de texte, analyse d'opinions, reconnaissance d'entités nommées etc.). Ces méthodes, beaucoup plus coûteuses que les méthodes d'évaluation intrinsèque, évaluent les embeddings en tant que composant d'un système et ne donnent pas d'information sur la qualité des embeddings eux-mêmes (Schnabel *et al.*, 2015).

Plusieurs travaux récents se penchent également sur la structure globale des espaces vectoriels produits par ces méthodes, comme Trost & Klakow (2017) afin de mettre au jour les différents biais pouvant exister dans les représentations des différents mots du lexique. Du point de vue de la linguistique de corpus, une autre façon intéressante de comparer deux modèles est celle présentée entre autres par Hamilton *et al.* (2016) qui cherchent les variations pour notamment observer les évolutions des représentations dans un corpus diachronique. Enfin, plus récemment et plus proche du travail présenté ici, certaines études se sont concentrées sur l'étude plus précise de la variation entre des modèles distributionnels : Antoniak & Mimno (2018) ont étudié à la fois la stabilité d'un même modèle à travers des changement de taille, de structure et de nature des corpus utilisés mais aussi à travers ses variations aléatoires inhérentes.

2.2 Reproductibilité des embeddings

L'instabilité interne des embeddings est un phénomène globalement ignoré dans les diverses études qui les exploitent ou les explorent. Elle est dûe à plusieurs phases de l'algorithme qui font appel à des processus aléatoires. Les réseaux neuronaux, notamment, sur lesquels se basent les méthodes comme

Word2vec, ont une phase de détermination aléatoire des poids initiaux des connections entre les neurones, qui sont ensuite ajustés en fonction des données fournies durant l'apprentissage. Toujours dans le cas de *Word2vec*, la méthode d'échantillonnage négatif (*negative sampling*) implique de plus la génération aléatoire de paires mot/contexte. Notons que cette présence de l'aléatoire dans la construction de modèles distributionnels n'est pas l'apanage des seuls réseaux de neurones, puisque par exemple la réduction de dimensions couramment utilisée dans les modèles fréquentiels est souvent faite avec des principes similaires, notamment les variantes randomisées de la décomposition en valeurs singulières (Sahlgren, 2005).

Cette instabilité a été soulevée par Hellrich & Hahn (2016) qui ont mesuré plus précisément les variations des premiers voisins des mots à travers des ensembles de 3 modèles *Word2vec* entraînés avec des hyperparamètres identiques. Ils ont notamment étudié le rôle de la fréquence des mots et surtout du nombre d'itérations dans la stabilité de ces voisinages. Plus récemment, Antoniak & Mimno (2018) ont montré la grande variation des voisinages sémantiques obtenus par plusieurs techniques de construction des embeddings et leur sensibilité à des variations minimes des données (comme l'ordre des documents dans le corpus), ainsi qu'au rôle des processus aléatoires impliqués. Ces deux études mettent en garde les utilisateurs de ces outils sur la nécessité de répéter plusieurs entraînements et d'étudier leur divergence avant de conclure, ce qui a bien entendu un coût calculatoire élevé. Notre approche diffère de ces deux études par le fait que nous cherchons plus précisément à étudier comment cette variation se distribue à travers le lexique, et plus précisément à caractériser en quoi certains mots ou ensemble de mots sont plus ou moins affectés par les processus aléatoires.

La réplicabilité (répétition à l'identique) et la reproductibilité (répétition en faisant varier un des paramètres, généralement les données) des expérimentations sont des questions de fond dans toutes les sciences expérimentales (on parle de 'reproducibility crisis') et plus récemment en informatique au premier rang de laquelle on trouve les travaux basés sur de l'apprentissage automatique (Hutson, 2018). En témoigne l'organisation récente de conférences et d'ateliers portant spécifiquement sur la question de la reproductibilité de travaux précédents. [1]

Une des façons de contrôler les processus aléatoires consiste à fixer la graine (*seed*) du générateur de nombres aléatoires, (cf. la règle numéro 6 des bonnes pratiques proposées par Sandve *et al.* (2013)), mais on conviendra que cette solution n'est pas satisfaisante intellectuellement, même si elle a l'intérêt de permettre la réplicabilité d'une expérience.

La reproductibilité est une question philosophique qui dépasse notre propos ici et qui nous amènerait à distinguer plus précisément ce qui relève, dans toute manipulation de données, d'un choix informé dans un processus déterministe (le seuil de fréquence, tel ou tel lemmatiseur, supprime-t-on ou non telle catégorie de mots-outils, etc.), de la sélection d'une valeur pour un hyperparamètre aux conséquences mal maîtrisées (nombre de dimensions, d'itérations, taux d'échantillonnage, etc.) et enfin de ce qui n'a simplement pas de sens ni de conséquence prévisible (la graine du générateur de nombres aléatoires). Plus modestement, prendre la mesure de ce phénomène simple et inévitable qu'est la part d'aléatoire dans le processus de construction des embeddings est pour nous une façon d'aborder la prise en main de ces techniques. Notre postulat est qu'un résultat affecté largement par une instabilité dûe aux seuls facteurs véritablement aléatoires est de moindre valeur qu'un phénomène apparaissant de façon répétée à travers des expériences similaires. Partant de là, notre but est de mieux cerner ce qui, pour le cas des embeddings, nous permet de mieux comprendre ce qui est à la portée de ces techniques, et à terme de délimiter les choix sur lesquels on peut véritablement agir en

1. Citons l'atelier *Reproducibility in Machine Learning* à ICML'17 ou encore les tâches participatives proposées par CENTRE (*CLEF, NTCIR, TREC Reproducibility*) à CLEF 2018.

connaissance de cause. Un autre point qui nous distingue des études sur l'instabilité présentées plus haut est que nous ne cherchons pas à contourner celle-ci, mais plutôt à la considérer comme un indice pour une meilleure compréhension de ces méthodes.

3 Dispositif

Nous résumons ici le dispositif expérimental mis en place, à savoir les modèles que nous avons construits et comment nous les comparons.

3.1 Modèles

Nous avons sélectionné une configuration unique en utilisant *Word2vec* avec les paramètres par défaut (code source original, méthode *skip-gram* avec *negative sampling* (taux de 5), fenêtre de taille 5, vecteurs de dimension 100, sous-échantillonnage des mots de fréquence supérieure à 10^{-3}, 5 itérations). Nous l'avons appliquée à un corpus générique de taille moyenne, le BNC (100 millions de mots, lemmatisés et catégorisés par Talismane (Urieli & Tanguy, 2013)), avec un seuil de fréquence minimale de 100. Cette même configuration a été utilisée 5 fois afin d'en tester la stabilité en comparant deux à deux les modèles produits. Ces modèles ne diffèrent donc que par l'effet des processus aléatoires inhérents à l'algorithme de *Word2vec* (voir § 2.2).

Nous avons répété l'opération en utilisant le même outil avec les mêmes paramètres sur deux autres corpus de taille similaire mais de nature très différente. Le premier corpus est le *ACL Anthology Reference corpus* (Bird *et al.*, 2008), constitué d'articles scientifiques dans le domaine du TAL, et le second est un corpus également constitué d'articles scientifiques, mais en biologie, constitué à partir du corpus *All of PLOS*.[2] Ces deux corpus ont eux aussi une taille de 100 millions de mots, et ont été traités de la même façon que le BNC.

3.2 Mesure de la variation

Pour estimer la variation entre deux modèles nous reprenons la méthode déjà utilisée par Sahlgren (2006) et reprise plus récemment par Antoniak & Mimno (2018) qui consiste à mesurer le recouvrement des N mots les plus similaires (plus proches voisins) d'un même mot-cible. Cette comparaison se fait indépendamment de l'ordre des voisins, et utilise généralement une valeur de N limitée. La formule ci-dessous indique comment est calculé le taux de variation (compris entre 0 et 1) pour un mot m entre deux modèles M_1 et M_2, $vois^N_M(m)$ représentant l'ensemble des N mots les plus similaires à m dans le modèle M.

$$var^N_{M_1,M_2}(m) = 1 - \frac{|vois^N_{M_1}(m) \cap vois^N_{M_2}(m)|}{N}$$

La mesure que nous avois choisie pour calculer les plus proches voisins est la similarité cosinus, désormais reconnue comme étant celle qui donne les meilleurs résultats dans l'exploitation des espaces distributionnels.

2. https://www.plos.org/text-and-data-mining

Cette mesure de la variation a bien entendu un ensemble d'avantages et d'inconvénients. Pour ce qui est des avantages, notons :

— La simplicité et la transparence : le taux de recouvrement est facilement interprétable, puisqu'il donne directement une estimation du nombre de voisins distributionnels que l'on retrouve d'un modèle à l'autre ;

— La facilité de calcul : l'extraction des plus proches voisins d'un mot est une procédure standard et bien optimisée dans les techniques d'exploitation des embeddings ;

— La localité : cette mesure permet d'attribuer un score de variation à chaque mot de l'espace sémantique et permet donc de comparer leur stabilité. Il est toutefois également possible d'utiliser une moyenne sur l'ensemble des mots pour obtenir une mesure globale de la variation entre deux modèles ;

— La pertinence : l'examen des plus proches voisins d'un mot est la façon la plus classique d'observer un espace sémantique, et il s'agit notamment de la procédure qui permet l'utilisation des espaces distributionnels en linguistique de corpus.

Les inconvénients principaux sont :

— L'absence de prise en compte de l'ordre des voisins : le fait de considérer les voisins non ordonnés ne donne qu'une image partielle de la variation locale entre deux modèles ;

— La sensibilité aux phénomènes de *hubness* : il est bien connu que la proximité dans les espaces vectoriels de haute dimension présente des biais que l'on qualifie de malédiction de la dimensionnalité (*dimensionality curse*) et notamment que certains points particuliers (*hubs*) ont tendance à apparaître très fréquemment dans les voisinages proches (Radovanović *et al.*, 2010). Ces phénomènes complexes peuvent naturellement perturber l'estimation de la stabilité ainsi mesurée.

— La nécessité de fixer N : le nombre de voisins considérés pour mesurer le recouvrement est bien entendu un paramètre très important qu'il est nécessaire de fixer a priori.

Nous comptons bien entendu aborder les questions soulevées par ces trois inconvénients, mais nous nous limitons dans cet article au dernier point, comme nous le verrons dans la section suivante.

4 Vue d'ensemble de la variation

Dans cette section nous présentons les mesures globales de la variation entre les différents modèles que nous avons entraînés, en considérant la variation moyenne observée sur l'ensemble des mots, mais également en mesurant la stabilité d'un échantillon de valeurs de similarité. La première question à traiter concerne le choix d'une valeur pour N, le nombre de plus proches voisins considérés.

4.1 Impact du nombre de voisins

Nous avons mesuré la variation moyenne pour chaque mot (sur les 10 paires de modèles, mais en traitant chaque corpus séparément) en utilisant différentes valeurs de N (1, 5, 10, 25, 50 et 100). Comme on peut le voir sur la figure 1 les scores moyens sont autour de 0.2, avec un faible écart-type autour des valeurs moyennes (indiqué par les deux traits pour chaque point de mesure) et décroissent naturellement lorsque N augmente tout en se stabilisant rapidement. La forme de cette évolution et le niveau de variation sont similaires sur les trois corpus.

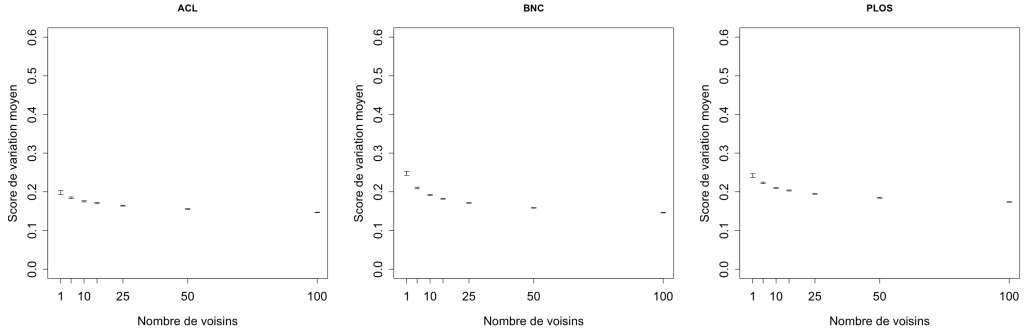


FIGURE 1. Variation moyenne pour différentes valeurs de N

Hellrich & Hahn (2016) ont observé la même stabilisation de leur score suivant le nombre de voisins considérés, mais ont opté pour un choix de N=1. Ce choix ne nous paraissait pas idéal pour l'observation plus détaillée des voisinages et nous avons pour notre part fixé N à 25. Cette valeur a été déterminée de la façon suivante. Pour les modèles entraînés sur le corpus BNC, nous avons calculé les coefficients de corrélation (sur l'ensemble des mots) entre les scores de variation obtenus pour toutes les paires possibles de valeurs de N. En faisant la moyenne de ces coefficients, nous avons observé que les valeurs obtenues pour N=25 maximisaient cette corrélation moyenne, autrement dit qu'elles étaient les plus représentatives des autres valeurs considérées pour N. Toutes les mesures présentées dans la suite de l'article utilisent donc cette valeur de N.

4.2 Variation moyenne

Pour chacun de nos trois corpus, nous avons mesuré la variation (sur les 25 premiers voisins, comme indiqué précédemment) entre chaque paire de modèles (5 modèles, 10 paires) et pour chaque mot du corpus atteignant le seuil minimal de 100 occurrences mais en nous limitant pour la suite aux seules classes ouvertes (adjectifs, adverbes, substantifs, noms propres et verbes). La taille du vocabulaire ainsi sélectionné dans chaque corpus est indiquée en table 2, avec les scores moyens de variation, l'écart-type moyen mesuré sur les 10 paires de modèles et enfin l'écart-type de la variation moyenne mesurée pour chaque mot.

Corpus	Vocabulaire	Var_{25} moyen	Moyenne de l'écart type	Écart-type des moyennes
ACL	22 292	0,16	0,04	0,08
BNC	27 434	0,17	0,04	0,07
PLOS	31 529	0,18	0,05	0,07

TABLE 2. Taux de variation global pour chaque corpus, calculé sur l'ensemble du vocabulaire et sur les 10 paires de modèles comparés

Globalement on observe un taux de variation moyen autour de 0,17. Rappelons que cela signifie que, lorsque l'on extrait les 25 mots les plus proches d'un mot-cible, 4 à 5 d'entre eux ne se retrouvent pas d'un modèle à l'autre. Ce taux semble stable à travers les 10 paires de modèles comparés pour chaque corpus (écart-type de 0,04 en moyenne).

Il existe par contre des différences plus importantes de cette même variation d'un mot à l'autre.

L'écart-type de la variation moyenne est environ de 0,07 (sur plus de 20 000 mots) : certains mots atteignent des scores de variation de 0,8, alors que d'autres ont des scores nuls (indiquant qu'ils ont les mêmes 25 premiers voisins dans tous les modèles, indépendamment de leur ordre).

De plus, il apparaît clairement que la variabilité est bien liée au mot et pas au modèle, puisqu'à travers les 10 paires de modèles comparés les scores de variation sont très stables : le coefficient de corrélation moyen (de Spearman, sur plus de 20 000 mots) entre les 10 paires est de 0,74 pour ACL, 0,72 pour le BNC et 0,77 pour PLOS. Autrement dit, il existe de façon inhérente des mots pour lesquels *Word2vec* va proposer les mêmes voisins dans chaque modèle (pour une configuration et un corpus donnés) et d'autres pour lesquels les voisins seront très sensibles aux aléas de la méthode.

4.3 Variation des scores d'évaluation sur les *benchmarks*

Bien que les méthodes d'évaluation interne classiques soient très critiquées (cf. section 2.1), notamment pour leur taille, l'imprécision des mesures humaines et leur faible taux d'accord inter-annotateur, nous avons voulu étudier l'instabilité de nos modèles face à celles-ci. Nous avons calculé le score de chacun de nos modèles sur les jeux de test WordSim353 (Finkelstein *et al.*, 2002) et Simlex-999 (Hill *et al.*, 2015). La table 3 donne les valeurs minimales et maximales obtenues pour nos trois séries de 5 modèles. La variation de ces scores d'un modèle à un autre sur un même corpus existe bien mais est relativement faible, de l'ordre de 1 à 4% du score (en relatif) et est très inférieure à la variation que l'on peut observer en comparant les modèles obtenus sur des corpus différents (où la variation va de 7 à 36%). Rappelons que de tels scores sont obtenus par la corrélation (des rangs) des scores de similarités avec ceux obtenus en interrogeant des humains sur quelques centaines de paires de mots.

Corpus	WordSim353 (min–max)	Simlex-999 (min–max)
ACL	0.592 – 0.601	0.192 – 0.201
BNC	0.631 – 0.639	0.306 – 0.312
PLOS	0.392 – 0.403	0.273 – 0.279

TABLE 3. Variation des scores sur deux jeux d'évaluation

Nous avons également profité du jeu de test WordSim353 pour regarder dans quelle mesure les scores de similarité (cosinus) entre deux mots varient d'un modèle à l'autre. Le choix de ne regarder que ces 353 paires de mots est dû au temps de calcul nécessaire pour comparer ces scores sur l'ensemble des mots du corpus (près d'un milliard de paires de mots par corpus). Nous avons observé une variation moyenne de ces scores de l'ordre de 4% de leur valeur d'un modèle à un autre, donc légèrement plus importante en valeur relative que les scores globaux du banc de test. Mais là aussi on observe des différences entre les items du jeu de test, sans d'ailleurs que l'ampleur de la variation soit liée à la proximité estimée par les humains entre les deux mots.

Enfin, il se trouve que le vocabulaire utilisé dans ces jeux de test a un taux de variation significativement plus faible que le reste (0.16 vs 0.17, test de Student, $p<0,05$, avec une variation de 1 à 2% en absolu selon les corpus). Il est donc raisonnable d'estimer que les scores de cosinus de ces paires varierait légèrement moins que l'ensemble des autres. Dans tous les cas, il semble clair que ces jeux de test ne permettent pas de prendre en compte à sa juste mesure l'ampleur du phénomène de variabilité interne des modèles.

5 Exploration de la variation

Comme nous l'avons remarqué précédemment, la variation n'est pas homogène à travers le lexique, puisque certains mots ont un score de variation nul et pour certains autres la valeur monte jusqu'à 0.8. Nous avons donc procédé à différentes explorations de ces différences, en cherchant à identifier ce qui permettrait de distinguer les mots stables des mots instables. Dans cette dernière section, nous abordons donc tout d'abord la question en regardant si certaines caractéristiques simples des mots comme leur fréquence ou leur catégorie morphosyntaxique étaient corrélées à leur variation, puis en examinant plus précisément les deux extrémités du spectre de la variation interne.

5.1 Impact de la fréquence et de la catégorie

Nous avons tout d'abord regardé la variation du score de variation en fonction de la fréquence du mot dans le corpus et sa catégorie morphosyntaxique. Différentes études ont pu montrer que la fréquence d'un mot dans un corpus influence la qualité de sa représentation dans les espaces distributionnels et qu'on obtient de meilleurs résultats suivant les différentes méthodes d'évaluation pour les mots de haute fréquence et par corrolaire sur des corpus de grande taille (Sahlgren & Lenci, 2016).

La figure 2 montre le score de variation moyen obtenu sur nos trois corpus pour différentes classes de fréquence logarithmique. Si une tendance linéaire simple ne semble pas se dégager aussi clairement qu'attendu, on peut résumer le lien en indiquant que ce sont les mots de fréquence intermédiaire (entre 1000 et 10 000 occurrences) qui sont les plus stables. En-deçà ou au-delà de cette zone il y a une légère augmentation de la variation.

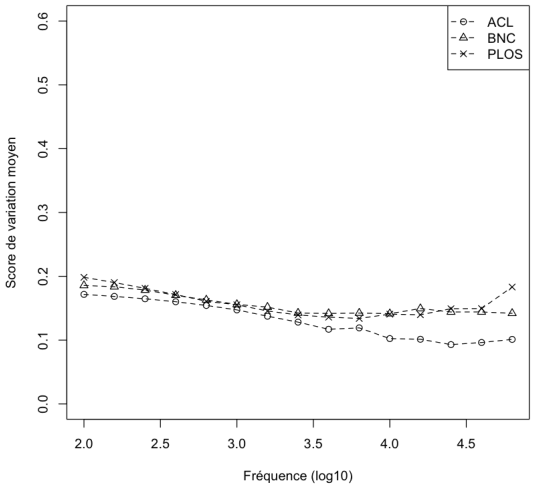


FIGURE 2. Effet de la fréquence sur le score moyen de variation pour ACL, BNC et PLOS

Pour ce qui est de la catégorie morphosyntaxique, on observe une grande homogénéité d'une partie du discours à l'autre. Seuls les noms propres ont une variation légèrement supérieure aux autres catégories, et ce pour les trois corpus.

Il semblerait donc que la variation soit à rechercher en lien avec des caractéristiques plus fines des mots. Pour cela, nous avons eu recours à une observation directe des mots stables et instables.

5.2 Clusters stables

A l'œil nu, en parcourant la liste ordonnée des mots les plus stables, on observe rapidement des régularités sous la forme de classes de mots apparemment similaires et ce, pour chacun des trois corpus (avec cependant des classes différentes d'un corpus à l'autre). C'est donc dans la direction du répérage de ces classes de mots avec une faible variation que nous avons orienté notre investigation.

Si l'on se penche sur le voisinage des mots, on observe plusieurs faits intéressants. Tout d'abord, le score de proximité du premier voisin n'est que partiellement corrélé avec le score de variation (-0,40 en moyenne). Cela signifie que le fait qu'un mot ait un premier voisin très proche explique en partie que son voisinage va rester stable. Il est en effet logique que ce proche voisin résiste aux aléas et qu'il fasse donc partie du voisinage stable de ce mot. En même temps, cela n'est absolument pas systématique, et on trouve de nombreux cas d'instabilité alors que des mots ont des voisins proches, et vice-versa. Ensuite, le score de variation d'un mot stable est corrélé avec la stabilité de ses voisins. Si l'on calcule en effet, lorsque l'on compare deux modèles, la variation moyenne des voisins (en faisant l'union des 25 plus proches voisins d'un mot dans chacun des deux modèles), on obtient une corrélation de 0,5 en moyenne avec le score de variation de ce mot. Cela confirme un mécanisme logique : la stabilité d'un mot entre deux modèles similaires est dûe à la présence de zones stables dans les espaces sémantiques. Il est donc possible d'identifier ces zones de stabilité.

En utilisant la variation mesurée pour chaque paire de modèles, nous avons calculé la variation globale moyenne d'un mot. Nous avons ensuite sélectionné pour chaque corpus les 300 mots montrant la variation moyenne la plus basse et pour chaque modèle entraîné, nous avons calculé leur similarité deux à deux dans chacun des cinq modèles. Nous avons appliqué une classification hiérarchique ascendante, en fixant le nombre de clusters à 10, et avons ainsi obtenu 5 partitions différentes. Pour tester la fiabilité de ces clusters, nous avons calculé l'indice de Rand (Rand, 1971) entre les 5 partitions générées (séparément pour chaque corpus). L'indice de Rand moyen était de 0,93 ($\pm$ 0,01, IC 95%) pour les 3 corpus, confirmant la grande stabilité des clusters extraits. Ceux que nous avons identifés et qualifiés sont indiqués dans la table 4.

On peut voir dans cette table des zones du lexique pour lesquelles on s'attend effectivement à une grande efficacité des méthodes distributionnelles, surtout lorsque les contextes sont restreints, spécifiques et réguliers. Les classes de co-hyponymes apparaissent clairement et répondent bien aux principes de base de la sémantique distributionnelle. Certaines classes enfin sont propres à nos corpus scientifiques comme le lexique transdiciplinaire, dont il a été démontré qu'il était bien capté par ces méthodes (Tutin, 2007). Il s'agit donc au final de zones lexicales classiques au sens de ce que permet la sémantique distributionnelle : on notera que ces clusters sont de taille variée, allant de quelques éléments pour les mesures de performance à plusieurs dizaines pour les ordinaux ou les références internes, ce qui nous permet d'envisager qu'ils ne sont pas dus au biais du nombre de voisins considérés.

5.3 Mots instables

A l'opposé du spectre de la variation, il est bien entendu beaucoup plus difficile d'identifier des régularités par une telle méthode. Puisque le principe même du clustering est inapproprié nous avons eu recours à une observation directe des 300 mots les plus instables et identifié plusieurs cas dont certains sont présentés en table 5.

Type de cluster	Corpus	Exemples
Contextes locaux spécifiques	ACL	mots de langues étrangères utilisés dans les exemples (*para, com, sobre...*), (*der, das, nicht, die...*) ordinaux (*12th, eleventh, 41st...*)
	PLOS	renvois internes (*figures, table, 6b, 1a...*) descriptions de figures (*dot, triangle, filled, orange...*)
	BNC	expressions temporelles (*am, pm, 31st, noon...*) mesures dans les recettes de cuisine (*tsp, tbsp, oz...*)
Classes fermées de co-hyponymes	ACL	mesures de performances (*precision, recall, f-score...*) traitements (*parsing, lemmatization, tokenizing...*)
	PLOS	antibiotiques (*puromycin, blasticidin, cefotaxime...*) voies d'administration (*intraperitoneally, intranasal, intramuscular...*)
	BNC	famille (*wife, grandmother, son, sister...*) pièces et objets de la maison (*kitchen, sitting-room, bathroom, furniture...*)
Phraséologie scientifique	ACL	adverbes de connection (*nevertheless, relatively, secondly, additionally...*) processus scientifique (*discuss, describe, observe...*)
	PLOS	adverbes de connection (*moreover, furthermore, conversely...*) processus scientifique (*hypothetize, reason, elucidate...*)

TABLE 4. Exemples de clusters stables identifiés pour chaque corpus

Parmi les mots les plus instables non reportés dans la table 5, nous avons également repéré des erreurs d'étiquetage, ainsi que des mots de basse fréquence. Cependant, les noms propres en général sont très présents parmi les mots les plus instables dans les trois corpus, qu'il s'agisse de patronymes (ACL et BNC) ou de sigles (PLOS). Pour les autres classes que nous proposons, on observe des génériques du domaine dont la fréquence est élevée et qui ont beaucoup d'hyponymes lointains mais pas de synonymes ni de voisins privilégiés. Nous observons également des adjectifs génériques ainsi que des mots très polysémiques, ces derniers étant plus difficiles à repérer, pour lesquels le mécanisme distributionnel peine effectivement à dégager des régularités dans les contextes d'emplois. Il est intéressant de noter toutefois que cela ne signifie pas que ces mots n'ont pas des voisins pertinents et globalement bien identifiés par les modèles, mais que ceux-ci sont noyés dans un environnement de très haute variation. Par exemple, un adjectif très instable comme *super* dans le BNC a bien des voisins sémantiquement pertinents à travers les différentes instances des modèles (comme *fabulous*,

Corpus	Séries	Exemples
ACL	noms propres	*Steve*, *Joyce*, *Ivan...*
	génériques du domaine	*language*, *sign*
	adjectifs génériques	*free*, *mix*, *special*
	mots polysémiques	*account*, *card*, *zone*, *sign*
PLOS	noms propres	*PCB*, *DMC*, *TLP*, *ACD ...*
	génériques du domaine	*gene*, *cell*, *protein*
	adjectifs génériques	*free*, *current*, *near*, *double*
BNC	noms propres	*Bart*, *Vince*, *Lewis...*
	adjectifs génériques	*whole*, *general*, *super*
	mots polysémiques	*make*, *close*, *cast*

TABLE 5. Exemples de classes de mots instables identifiées pour chaque corpus

stylish, *stunning* et quantité d'autres laudatifs) mais on y trouve en très bonne place de nombreux parasites extrêmement dispersés et variant d'un modèle à l'autre (comme *Granada*, *cracker*, *zeppelin*).

6 Conclusion et perspectives

Cette étude nous a permis de mesurer la stabilité interne des modèles produits par *Word2vec* dans son paramétrage par défaut. Globalement 17% des 25 mots les plus proches sont susceptibles d'être différents, uniquement à cause des facteurs aléatoires intervenant dans la méthode. Ce taux est toutefois inférieur pour certains clusters denses de mots pour lesquels le mécanisme distributionnel est très robuste. Par contre, certaines séries de mots semblent très sensibles à l'instabilité (noms propres, mots génériques du domaine). Cette instabilité inhérente est malheureusement souvent ignorée et est d'ailleurs minime quand on compare deux modèles sur des bancs de test classiques. À ce stade, nous n'avons pas estimé l'impact de cette variabilité sur des tâches externes de TAL qui se basent sur les embeddings (parsing, extraction d'information, classification de documents, etc.) mais il est probable que l'effet soit minime, ou même qu'il se perde dans la variabilité inhérente aux outils qui utilisent ces données et ont de grandes chances d'être soumis aux mêmes variations aléatoires.

Notre point de vue sur la question est plus orienté vers l'utilisation des embeddings comme outil d'exploration sémantique de mots ou de corpus. C'est cette orientation qui nous a guidés vers la mesure de la variation des plus proches voisins, l'observation de ceux-ci étant le mode principal d'investigation de la représentation sémantique d'un mot dans de tels espaces vectoriels. Au vu de l'ampleur du phénomène, nous rejoignons les conclusions et consignes de Antoniak & Mimno (2018) sur les précautions avec lesquelles aborder les résultats. Si la solution la plus directe est de mutliplier les modèles avant de tirer des conclusions, nous espérons toutefois parvenir à une procédure d'identification de ce qui pourrait expliquer la stabilité relative de la représentation d'un mot, et par là même prédire à terme la fiabilité de celle-ci. De premiers résultats encourageants dans ce sens sont présentés dans (Pierrejean & Tanguy, 2018), mais de nombreuses questions restent à ce stade en suspens, notamment concernant le rôle des différents facteurs sur cette variabilité (taille du corpus et hyperparamètres), ainsi que l'utilisation d'une mesure plus fine de la variation qui prendrait l'ordre des voisins en considération.

Au-delà de cette instabilité "interne" des modèles, notre objectif est également de nous pencher sur la variation observable entre des modèles différents et au premier chef lorsque l'on passe d'un corpus à l'autre, à l'instar de ce qu'ont fait Hamilton *et al.* (2016) en étudiant les changements diachroniques. C'est cet objectif initial qui nous avait fait choisir des corpus différents comme ACL et PLOS. Il est bien entendu possible de mesurer par le même recouvrement quels sont les mots (ou classes) dont la représentation change le plus. Toutefois, on ne peut aborder ces questions sans avoir au préalable une estimation de la part du hasard dans chacun des modèles. Plus généralement, c'est une meilleure compréhension des mécanismes et des limites de ces outils qui permettra de les intégrer dans des travaux d'investigation plus fins et plus fiables en linguistique de corpus. Si nous reconnaissons comme bien d'autres leur spectaculaire efficacité et leur facilité d'utilisation, elles ne doivent pas nous en masquer les limites et les défauts.

Remerciements

Les expériences présentées dans cet article ont été réalisées en utilisant la plateforme OSIRIM administrée par l'IRIT et soutenue par le CNRS, la région Midi-Pyrénées, le gouvernement français, et le FEDER (voir `http://osirim.irit.fr/site/fr`).

Références

ANTONIAK M. & MIMNO D. (2018). Evaluating the Stability of Embedding-based Word Similarities. *Transactions of the Association for Computational Linguistics*, **6**, 107–119.

ASR F. T., WILLITS J. A. & JONES M. N. (2016). Comparing Predictive and Co-occurrence Based Models of Lexical Semantics Trained on Child-directed Speech. In *Proceedings of the 37th Meeting of the Cognitive Science Society*.

BERNIER-COLBORNE G. & DROUIN P. (2016). Evaluation of distributional semantic models : a holistic approach. In *Proceedings of the 5th International Workshop on Computational Terminology*, p. 52–61, Osaka, Japan.

BIRD S., DALE R., DORR B., GIBSON B., JOSEPH M., KAN M.-Y., LEE D., POWLEY B., RADEV D. & FAN TAN Y. (2008). The ACL Anthology Reference Corpus : A Reference Dataset for Bibliographic Research in Computational Linguistics. In *Proceedings of Language Resources and Evaluation Conference (LREC 08)*, Marrakesh, Morrocco.

CHIU B., CRICHTON G., KORHONEN A. & PYYSALO S. (2016). How to Train Good Word Embeddings for Biomedical NLP. In *Proceedings of the 15th Workshop on Biomedical Natural Language Processing*, p. 166–174, Berlin, Germany.

FARUQUI M., TSVETKOV Y., RASTOGI P. & DYER C. (2016). Problems with evaluation of word embeddings using word similarity tasks. In *Proceedings of the 1st Workshop on Evaluating Vector-Space Representations for NLP*, p. 30–35.

FINKELSTEIN L., GABRILOVICH E., MATIAS Y., RIVLIN E., SOLAN Z., WOLFMAN G. & RUPPIN E. (2002). Placing Search in Context : The Concept Revisited. *ACM Transactions on Information Systems*, **20**(1), 116–131.

GOLDBERG Y. (2016). A Primer on Neural Network Models for Natural Language Processing. *Journal of Artificial Intelligence Research*, **57**, 345–420.

HAMILTON W. L., LESKOVEC J. & JURAFSKY D. (2016). Diachronic Word Embeddings Reveal Statistical Laws of Semantic Change. *Proceedings of ACL 2016*, p. 1489–1501.

HELLRICH J. & HAHN U. (2016). Bad Company - Neighborhoods in Neural Embedding Spaces Considered Harmful. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics : Technical Papers*, p. 2785–2796, Osaka, Japan.

HILL F., REICHART R. & KORHONEN A. (2015). SimLex-999 : Evaluating Semantic Models with (Genuine) Similarity Estimation. *Computational Linguistics*, **41**, 665–695.

HUTSON M. (2018). Artificial intelligence faces reproducibility crisis. *Science*, **359**(6377), 725–726.

LEVY O. & GOLDBERG Y. (2014). Dependency-Based Word Embeddings. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics*, p. 302–308, Baltimore, Maryland, USA.

LI B., LIU T., ZHAO Z., TANG B., DROZD A., ROGERS A. & DU X. (2017). Investigating Different Syntactic Context Types and Context Representations for Learning Word Embeddings. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, p. 2411–2421.

MELAMUD O., MCCLOSKY D., PATWARDHAN S. & BANSAL M. (2016). The Role of Context Types and Dimensionality in Learning Word Embeddings. In *Proceedings of NAACL-HLT 2016*, San Diego, California.

MIKOLOV T., CHEN K., CORRADO G. & DEAN J. (2013). Efficient Estimation of Word Representations in Vector Space. *CoRR*, **abs/1301.3781**.

PIERREJEAN B. & TANGUY L. (2018). Predicting word embeddings variability. In *Proceedings of the 7th Joint Conference on Lexical and Computational Semantics (*SEM 2018)*. To appear.

RADOVANOVIĆ M., NANOPOULOS A. & IVANOVIĆ M. (2010). Hubs in Space : Popular Nearest Neighbors in High-Dimensional Data. *Journal of Machine Learning Research*, **11**, 2487–2531.

RAND W. M. (1971). Objective Criteria for the Evaluation of Clustering Methods. *Journal of the American Statistical Association*, **66**(336), 846–850.

SAHLGREN M. (2005). An introduction to random indexing. In *Proceedings of the Methods and Applications of Semantic Indexing Workshop at the 7th International Conference on Terminology and Knowledge Engineering (TKE)*.

SAHLGREN M. (2006). *The Word-Space Model*. PhD thesis, Gothenburg University.

SAHLGREN M. & LENCI A. (2016). The Effects of Data Size and Frequency Range on Distributional Semantic Models. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, p. 975–980, Austin, Texas.

SANDVE G. K., NEKRUTENKO A., TAYLOR J. & HOVIG E. (2013). Ten simple rules for reproducible computational research. *PLoS computational biology*, **9**(10).

SCHNABEL T., LABUTOV I., MIMNO D. & JOACHIMS T. (2015). Evaluation methods for unsupervised word embeddings. *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, p. 298–307.

TROST T. A. & KLAKOW D. (2017). Parameter free hierarchical graph-based clustering for analyzing continuous word embeddings. In *Proceedings of TextGraphs-11 : the Workshop on Graph-based Methods for Natural Language Processing*.

TUTIN A. (2007). Traitement sémantique par analyse distributionnelle des noms transdisciplinaires des écrits scientifiques. In *Actes de la 14e conférence sur le Traitement Automatique des Langues Naturelles (TALN'2007)*, p. 283–292, Toulouse, France.

URIELI A. & TANGUY L. (2013). L'apport du faisceau dans l'analyse syntaxique en dépendances par transitions : études de cas avec l'analyseur Talismane. In *Actes de la 20e conférence sur le Traitement Automatique des Langues Naturelles (TALN'2013)*, p. 188–201, Les Sables d'Olonne, France.

Modeling infant segmentation of two morphologically diverse languages

Georgia Rengina Loukatou[1] Sabine Stoll [2] Damian Blasi[2] Alejandrina Cristia[1]
(1) LSCP, Département d'études cognitives, ENS, EHESS, CNRS, PSL Research University, Paris, France
(2) University of Zurich, Zurich, Switzerland
georgia.loukatou@ens.fr

RÉSUMÉ

Les nourrissons doivent trouver des limites de mots dans le flux continu de la parole. De nombreuses études computationnelles étudient de tels mécanismes. Cependant, la majorité d'entre elles se sont concentrées sur l'anglais, une langue morphologiquement simple et qui rend la tâche de segmentation aisée. Les langues polysynthétiques - pour lesquelles chaque mot est composé de plusieurs morphèmes - peuvent présenter des difficultés supplémentaires lors de la segmentation. De plus, le mot est considéré comme la cible de la segmentation, mais il est possible que les nourrissons segmentent des morphèmes et non pas des mots. Notre étude se concentre sur deux langues ayant des structures morphologiques différentes, le chintang et le japonais. Trois algorithmes de segmentation conceptuellement variés sont évalués sur des représentations de mots et de morphèmes. L'évaluation de ces algorithmes nous mène à tirer plusieurs conclusions. Le modèle lexical est le plus performant, notamment lorsqu'on considère les morphèmes et non pas les mots. De plus, en faisant varier leur évaluation en fonction de la langue, le japonais nous apporte de meilleurs résultats.

ABSTRACT

A rich literature explores unsupervised segmentation algorithms infants could use to parse their input, mainly focusing on English, an analytic language where word, morpheme, and syllable boundaries often coincide. Synthetic languages, where words are multi-morphemic, may present unique difficulties for segmentation. Our study tests corpora of two languages selected to differ in the extent of complexity of their morphological structure, Chintang and Japanese. We use three conceptually diverse word segmentation algorithms and we evaluate them on both word- and morpheme-level representations. As predicted, results for the simpler Japanese are better than those for the more complex Chintang. However, the difference is small compared to the effect of the algorithm (with the lexical algorithm outperforming sub-lexical ones) and the level (scores were lower when evaluating on words versus morphemes). There are also important interactions between language, model, and evaluation level, which ought to be considered in future work.

MOTS-CLÉS : variation interlinguistique, apprentissage statistique, segmentation des mots, acquisition du langage.

KEYWORDS: cross-linguistic variation, statistical learning, word segmentation, language acquisition.

1 Introduction

Human infants are known to acquire a comprehension vocabulary of hundreds of words by two years of age (Hoff, 2013), and probably accumulate a protolexicon consisting solely of word forms, with no meaning attached, by the end of the first year (Ngon *et al.*, 2013). Infants' discovery of basic units in their input has been modeled as follows. A transcript of infant-directed speech is converted into phonological text, word boundaries are removed, and an algorithm is applied. Then, the word boundaries posited by the algorithm (and the resulting word tokens) are compared against the adult segmentation found in the original corpus.

By and large, two classes of algorithms have been heavily studied (Daland, 2009; Jarosz & Johnson, 2013). Algorithms in the *lexical* class are built to find the most economical system of minimal units needed to reproduce the input. Those in the *sub-lexical* class aim to find local cues allowing the learner to posit boundaries, for instance detectable via a dip in transitional probabilities. We will discuss both in more detail below, but for now it suffices to say that both are plausible given known infant experimental data (Mersad & Nazzi, 2012; Saffran *et al.*, 1996a).

The present study represents the first systematic attempt to apply both types of unsupervised word form discovery techniques to two morphologically diverse languages. In the next section, we summarize previous work with the lens of morphological distinctions.

1.1 Morphological variability predicts performance in previous modeling work

Most previous work modeling infant segmentation has focused on English (e.g. Lignos 2011; Venkataraman 2001; Christiansen & Curtin 2005; Phillips & Pearl 2015; Monaghan & Christiansen 2010); yet this is not an "average" language for segmentation. English words are mostly monomorphemic and monosyllabic, such that word, morpheme, and syllable boundaries usually coincide (DeKeyser, 2005).

Indeed, morphologically speaking, English can be classified as an analytic language, because most words have few or no morphemes other than the root. Synthetic languages are characterized by having richer inflectional morphology. They use morphemes such as prefixes, suffixes and infixes to convey certain features (e.g., gender) and/or the relation between words in a sentence (e.g., via case).

Languages can vary greatly in the degree of synthesis. Some languages, such as Hungarian and Tamil, are synthetic to a high degree, with a rich inflectional morphology in both nouns and verbs. Others are more intermediate, including many IndoEuropean languages, such as Italian, Spanish, and French, which have only a few suffixes in verbs and nouns. A distinction among synthetic languages that we will not study but is worth mentioning is that between agglutinative and fusional languages. In the former, morphemes are transparent and concatenated, whereas in fusional languages a single morpheme contains many features.

Languages with a rich morphology are of particular interest in the context of segmentation. Since complex words are formed by the combination of many easily separable morphemes, lexical algorithms could break words up into the component morphemes (Batchelder, 2002). Additionally, highly synthetic languages usually have longer words, and may have longer utterances (in number of phonemes or syllables). Longer utterances mean more alternative parses can be posited, and thus more

uncertainty particularly (but not only) for lexical algorithms (Fourtassi *et al.*, 2013). Moreover, lexical algorithms often implement a drive for economy, whereby reuse of minimal units is preferred over postulation of additional lexical units. This could specifically lead to problems for languages where, by virtue of inflectional morphology, a lexeme has many surface forms, each used less frequently, and where it may be more economical to break up the word into roots and affixes, which can be more efficiently re-used.

Finally, corpora of such languages could contain fewer repetitions of each word token (since each lexeme can have different surface forms) and thus a higher proportion of hapaxes than analytic languages, which might affect performance in lexical algorithms where the probability of generating a word is partially a function of its frequency. All the above predict better performance for less than more synthetic languages, and for this difference to be more marked in the performance of lexical than sublexical algorithms.

Overall, previous results support our main predictions, with synthetic languages yielding lower segmentation performance than analytic languages. Fourtassi *et al.* (2013) applied a probabilistic lexicon-building algorithm on English and Japanese. English (analytic) yielded a Token F-score of 0.77 and Japanese (synthetic-agglutinative) 0.69. Qualitative inspection suggested to the authors that the algorithm broke apart morphological affixes, generating more oversegmentation errors for Japanese than English, which fits the reasoning laid out above well.

This effect was replicated by Boruta *et al.* (2011) with another lexicon-based model, documenting better results for English than French (synthetic-fusional), and for French than Japanese. The author reported a higher proportion of hapax words in Japanese than for English, and a lower likelihood of correct identification by the algorithm for hapaxes than words with more than one repetition. Finally, results were dismal for Sesotho, another highly synthetic language characterized by even more complex morphology than Japanese (Johnson, 2008).

Although the arguments above are most relevant to lexical algorithms, previous work using sub-lexical ones also confirms the hypothesized trend. A diphone-based segmentation model developed by Daland performed lower for the morphologically complex Russian than English. For their part, Saksida *et al.* (2017) used a set of segmentation models based on transitional probabilities on a range of corpora. English had a maximum score of 0.85, whereas Japanese, Tamil, and Hungarian, all synthetic, a maximum of 0.75.

As in the lexical literature, other work even suggests differences among synthetic languages. In Gervain & Erra (2012), better results were found for the less complex Italian than the more complex Hungarian. The authors commented that there may be more oversegmentation in the latter language, as some of the erroneously segmented words were real morphemes, which is interesting given that this unsupervised algorithm has not been designed to be sensitive to lexical and morphological composition.

2 The present study

The key question motivating this study is whether languages that vary in morphological complexity differ in segmentability. To answer it, we looked for languages that were morphologically diverse, but for which there were closely matched and comparable corpora. Previous authors have often argued that lower performance for highly synthetic languages arises from oversegmentation, mainly based on

Language	Verb agr.	Split erg.	Compactness	Syncr.	V syn.	N syn.
Japanese	none	low	cumulative	none	low	1
Chintang	some	medium	distributive	some	high	3

TABLE 1 – Differences between the two languages according to (Bickel *et al.*, 2013). Verb Agr(eement), Split Ergativity of Case (proportion of ergative case alignments), Compactness, (Prevalence of) Syncr(etism), V(erbal) Syn(thesis), N(ominal) Syn(thesis).

qualitative inspection of results (but see Johnson 2008). To assess this question more systematically, we inquired whether performance varied as a function of the level of linguistic representation on which segmentation is evaluated. [1]

Our goal was not to test an exhaustive list of languages, which was not feasible at present. Instead, we opted to compare two language corpora which are part of the same database and have been transcribed using the same guidelines. Additional desiderata included that the morphological difference of these languages should be large and computationally assessed, and that other linguistic parameters such as syllable structure should be in similar levels.

All of these considerations led us to the ACQDIV database of linguistically diverse languages (Schikowski *et al.*, 2015). Languages in this database have been sorted based on clustering algorithms and according to several linguistic, typological variables. It may be worth pointing out that we did not test our models on English here, as previous literature has extensively presented results on various English corpora (some if it has been summarized above), and English did not exist in the chosen database.

Thus, to best complement previous work, we selected two morphologically diverse non-IndoEuropean languages present in the AcqDiv database, namely Japanese and Chintang (Bickel *et al.*, 2013). [2] Even though the languages belong to the same morphological category (synthetic agglutinative), they are very different in the degree of complexity.

Chintang is a polysynthetic language (i.e., having an extremely high ratio of morphemes per word), and thus has a more complex morphology than Japanese. It has higher verb and noun synthesis (number of categories expressed, word complexity – compare Paudyal 2015 for Chintang, and Kuno 1973; Tsujimura 2013 for Japanese), with up to 10 morphemes per word in Chintang. The languages also differ in that Chintang has distributive inflectional compactness (categories are expressed separately in distinct morphemes), whereas Japanese has cumulative compactness (grammatical categories are expressed cumulatively in fusioned affixes), which denotes the need for less morphemes than for Chintang, as we can see in Table 1. As per our desideratum above, the phonological complexity (phonemic inventory and syllabic structure) is similar across the two languages.

Most previous work used a single class of algorithms (but see (Ludusan *et al.*, 2017)). Although previous literature suggests that segmentation performance varies as a function of morphological type for both lexical and prelexical algorithms, our reasoning above predicts that effects of morphological complexity on segmentation should be stronger for lexical than prelexical algorithms. We therefore included algorithms of both types.

1. Note that it is not unreasonable to propose that infants segment morphemes, rather than words (Phillips & Pearl, 2014).
2. Chintang is a language of the Kiranti subgroup of the Sino-Tibetan language family spoken in Eastern Nepal by about 6,000 speakers.

3 Methods

The Chintang recordings took place 4h (cumulated during several sessions carried out within a week) per month, over 18 months, and involved 7 children aged between 6 months and 4 years and 4 months of age (Stoll *et al.*, 2016). For Japanese, recordings of 7 children aged 1 year 4 months, to 5 years 1 month, each lasting 40-70 minutes, were collected between once per week and once per month.

All child-directed and child-overheard speech had been carefully transcribed in a transparent orthography (as was the child's own speech, which was not analyzed here). We applied grapheme-to-phoneme rules to derive the phonological representation of utterances. After processing, the Chintang corpus contained 296,939 utterances, with an average of 2.7 words, 5.4 syllables, and 11.4 phones per utterance ; and the Japanese corpus 264,945 utterances with an average of 3 words, 5.7 syllables, 11.2 phones per utterance). All utterances where morpheme annotation was incomplete were removed, so the Japanese morpheme corpus after processing contained 85267 utterances with 2 morphemes, 3.2 syllables, 6.3 phones per utterance and the Chintang morpheme corpus contained 280319 utterances with 4.5 morphemes, 5.5 syllables, 11.4 phones per utterance.

To perform inferential statistics, each corpus was divided in ten equal subparts based on number of utterances. Within-sentence word boundaries were removed and fed to three models, which varied on the cognitive strategies applied, as follows. [3]

3.1 DiBS

For the sub-lexical Diphone Based Segmentation model (DiBS) (Daland, 2009; Daland & Pierrehumbert, 2011), segmentation decisions are based on the basis of diphone probabilities. A probability of word boundary ranging from 0 up to 1 is assigned to each diphone found within every utterance. For the present work, we are using one of the unsupervised version of DiBS called phrasal DiBS. [4] In this version, the algorithm treats phrase edges as a proxy for word edges given the unquestionable assumption that diphones spanning a phrase boundary are also spanning word breaks. It estimates two parameters from the corpus, to be fed into the formula 1.

$$p(\#|xy) = \frac{f(\# \wedge xy)}{f(xy)} \tag{1}$$

where $f(\# \wedge xy)$ is the number of [xy] sequences with a phrase boundary in the middle, and $f(xy)$ is the the number of [xy] sequences in any position (Daland, 2009). When this ratio is higher than a parameter called "probability of word boundary", then the system will make a hard decision that there is indeed a break. The probability of word boundary is calculated using Formula 2.

$$\frac{Nw - Nu}{Np - Nu} \tag{2}$$

where Nw stands for number of words, Nu number of utterances, and Np number of phones.

3. We actually used the defaults in the WordSeg package (Bernard *et al.*, submitted). For more information, visit https ://wordseg.readthedocs.io/en/latest/algorithms.html

4. Other versions of DiBS, not used for this paper, are Baseline DiBS (where the boundary in $f(\# \wedge xy$ is the true word boundary, and thus is fully supervised), and Lexical DiBS (which requires a small vocabulary to be provided by the programmer, and thus necessitates additional assumptions as to how the child learned $those$ words).

Notice that while Formula 1 requires nothing but a phone inventory and the phrase boundary location, Formula 2 makes reference to the gold number of words, and thus can be said to be supervised. Daland argues convincingly that this can be viewed as a convenient shortcut, rather than a design flaw in the algorithm. Indeed, we can imagine children being born with a parameter akin to the minimal word length requirement (McCarthy & Prince, 1986), or deducing them from other aspects of the language (length of the shortest utterances, distance between stressed syllables, etc.)

In its original implementation, this model required that the input be coded in phonemes. To allow direct comparison with previous work, we did not alter this requirement.

3.2 TP

The Transitional Probabilities (TP) family builds on the assumption that the transitional probability between adjacent syllables is lower at word boundaries than at word middles (Gervain & Erra, 2012; Saffran *et al.*, 1996b; Saksida *et al.*, 2017). Forward TP (FTP) is defined as

$$FTP(AB) = \frac{f(AB)}{f(A)} \tag{3}$$

where $f(AB)$ is the frequency of occurrence of the syllabic sequence AB and $f(A)$ is the frequency of occurrence of the syllable A.

Backward TPs (BTP) instead divides the product by the times the second syllable appears. $P(B)$ is the probability of occurrence of the syllable B.

$$BTP(AB) = \frac{f(AB)}{f(B)} \tag{4}$$

Notice that these formulas simply provide an indirect estimate of how likely B is given A (or B, in Formula 4) and thus are not sufficient to posit a boundary. In fact, the decision on whether to posit a boundary between two syllables can be taken with at least two different methods. The Absolute TP method (TPa) uses the average of the TPs over the sum of bigrams for the whole corpus as a threshold. We can have either Backward TPa (BTPa) and Forward TPa (FTPa).

The Relative TP (TPr) cuts words when the TP value of a bigram is weaker than the TP of the neighboring ones, so we have Backward TPr (BTPr) and Forward TPr (FTPr). For example, if AB is a bigram in a sequence of XABY, then a boundary would be posited if Equation 5 is true.

$$TP(XA) > TP(AB) < TP(BY) \tag{5}$$

This model's input is coded into syllables, so we syllabified the corpora using the Maximal Onset Principle, according to which a syllable's onset should be extended as much as possible, as long as it stays phonotactically legal (Bartlett *et al.*, 2009).

All previous work has analyzed corpora unitized at the level of the syllable, and not of phonemes, as input to this family of algorithm. Although a TP version with phonemes as basic units, rather than syllables, is cognitively possible, we preferred the latter in order to compare our results against others' using TP as it had been used in previous studies. Reviewers indicated this results in a loss

of comparability across models, given that the other two models are based on phonemes. If a reader feels the same way at this point, we would like to underline here that our research goal was neither to provide an exhaustive mapping of all models nor even to carry out specific comparisons across the models. Our main research question, as stated above, concerns the effects of $morphological$ differences across languages. The use of varied models (defined as the conjunction of input and processing decisions) allows us to assess whether patterns are observed $despitemodelvariation$, or whether certain patterns may be only obvious for subsets of models.

3.3 AGu

The third model, a member of the Adaptor Grammar (AG) family, adopts a lexical approach (Goldwater *et al.*, 2009; Johnson & Demuth, 2010), meaning that it tries to find patterns of sequence of units that repeat in the input and uses that lexicon to parse the input. It is technically a great deal more complex than either of the models just discussed. For reasons of space, we provide a mainly verbal explanation, and refer readers to Johnson *et al.* (2007) for a technical introduction to adaptor grammars, including mathematical formulae and general properties.

In a nutshell, we use here a Pitman-Yor Adaptor Grammar, which is a generalized verion of probabilistic context-free grammars (PCGF) (Johnson *et al.*, 2007). In context-free grammars, corpora are generated as a function of the repeated application of a set of rewrite rules, which, in the process of parsing/generating a corpus, are selected independently and at random. Contrastingly, in PCGFs, each rule is assigned a probability, and the probability of a given parse is the product of the probabilities of the rules that may have been invoked to generate the input text.

Briefly, a given Adaptor Grammar is described as a function of (N,T,R,θ,a/b) N is a finite set of nonterminal symbols (in our grammar below, Sentence, Word, Phoneme), T is a finite set of terminal symbols (in our grammar below, the actual phonemes of the language), R is a set of rewrite rules, θ is a probability distribution over the different rules, and a/b are concentration parameters governing re-use versus generation (as explained below).

For this study, the input was represented using phonemes, and we used a basic version of adaptor grammar assuming no dependencies between words (unigram). For this reason, we will refer to our implementation of the model in what follows as AGu. This is the simplest adaptor grammar one can imagine, and it presupposes only two assumptions – (a) Sentences are composed of one or more reusable words, and (b) Words are composed of one or more basic terminal units (phonemes). These assumptions are encoded into rules enumerated below.

1. Sentence $\rightarrow$ Word (Word)
2. Word* $\rightarrow$ Phoneme (Phoneme)
3. Phoneme $\rightarrow$ a
4. Phoneme $\rightarrow$ ...

The items on the left are non-terminals and those on the right may be lower-level non-terminals (as in rewrite rules 1-2) or terminals (as in the remaining rewrite rules). Items between parentheses are optional. Items with an asterisk can be generated *d*e novo, or they can be added as rewrite rules during parsing, and subsequently re-used throughout the corpus. Notice additionally that these rewrite rules are equivalent to creating trees – for instance, the word "see" may be parsed as the application of rule (2) followed by the rules expanding the Phoneme non-terminal into the terminal phonemes /s/, /i/.

As just mentioned θ is a vector corresponding to each and every rule in R, a number that represents the probability of expanding the non-terminal on the left hand side of the given rule into possible terminal(s) or non-terminal(s) on the right hand side. Let us imagine a parse where the system finds a sequence A, which is a non-terminal. The Gaussian distribution G_A corresponds to the set of trees associated with the non-terminal A. For example, /sit/ could in theory be parsed as /s i/ or /si/ – G_A would establish that there is a probability p that corresponds to the tree resulting in the parse /s i/ with and $1 - p$ to the tree resulting in the parse /si/.

Recall that the system can parse the input using just the list of original rules, in which case the generation of an utterance results from the repeated application of only rules 1-2 and the terminal phonemes in the utterance. It can also, however, create new rules to shortcut this process. In the implementation we use for this paper, we allowed the creation of rules as sequences of phonemes, which effectively means we allowed the creation of a lexicon or morphological inventory, which can be used to segment utterances into a sequence of words or morphemes (without using the 'Phoneme $\rightarrow$ terminal' rules). In the version we are using (Johnson *et al.*, 2007), probabilities for re-use versus regeneration are based on the Pitman-Yor Process, a stochastic process of probability distribution which pits the reuse of frequently occurring trees (or rules) versus creating new trees (or rules). [5]

Specifically, this process is governed by the "concentration" set of parameters a and b, which determine whether generated rules are costly and thus whether the system should be reusing rules or create many new rules. For the sake of comparability with previous work, we used the values that had been preferred for experiments on English, French and Japanese adult and child corpora at the time the package was first used (e.g., Fourtassi *et al.* 2013), namely a=0.0001 and b=10000. To put this in context, initially, the lexicon is empty (i.e., no shortcut rules have been created), so the first utterance will be parsed as a sequence of words, each composed by phonemes, with this whole probability distribution governed by G_A. For simplicity, assume the system posits a single word – w_1. The second time the same sequence of terminals is found, the system can extract this word from the adaptor's lexicon (and hence it is w_1), generate another rule with the same non-terminal, or generate the word from scratch using G_A. This whole process is repeated in several runs of 2000 times. Finally, Minimum Bayes Risk is used to find the most common sample segmentations (Johnson *et al.*, 2007).

4 Results

Recall that our main research question is whether there are differences in segmentability across languages as a function of morphological complexity, particularly as the more complex language (Chintang) may be oversegmented when the output is evaluated at the level of words (rather than morphemes) and when using a lexical algorithm (AGu, as opposed to the sublexical DiBS and TPs). To answer this question, we fit a regression model to token F-scores data declaring language, level, algorithm and their interactions as independent predictors, and corpus as repeated variable (given that the corpora had been cut into 10 subparts so as to be able to carry out inferential statistics). This regression accounted for most of the variance in the data, $R^2 = .90$ ($F(23, 216) = 98.5$, $p < .001$).

As predicted, the coefficient estimating the language effect between Chintang and Japanese is positive (0.069), suggesting higher scores for the latter. Interestingly, this effect is smaller than that of level (morpheme versus word, -0.11). The presence of all 2- and 3-way interactions, however, discourages

5. It is often described as the Chinese restaurant process, where new customers can be seated at a new table or an extant table, and in the latter case they will tend to be placed in tables with many customers than tables with a few.

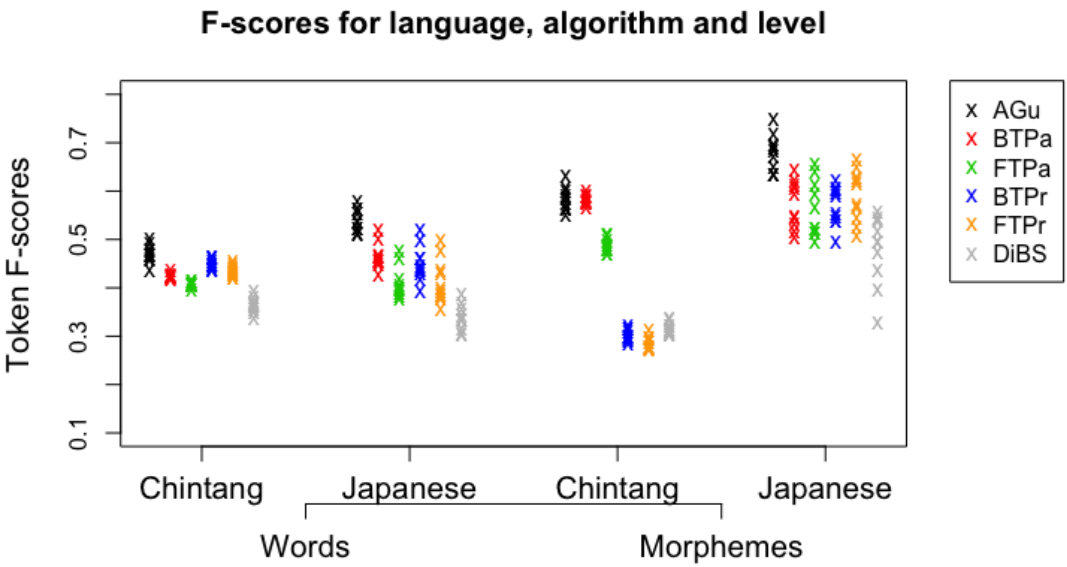


FIGURE 1 – Token F-scores across language and representational level. Models are marked by color. Each "x" represents one of the ten subparts of each corpus.

a simple reading of these main effects.

As clear in Figure 1 there were strong interactions between all three factors (language, level, model).

5 Discussion

When combined with results from the previous literature, we confirm the generalization that complex languages (such as the two studied here) lead to lower segmentation scores than morphologically impoverished languages (such as English). Indeed, for all algorithms, we obtained lower scores for Chintang and Japanese than English results previously documented. We focus on the word level for this comparison, since previous work has systematically evaluated performance on this level, and not the morpheme level. For AGu, we retrieve an average Token F-score of .54 for Japanese and .47 for Chintang, whose performance is close to that for Sesotho (Johnson, 2008), another morphologically complex language with high degrees of synthesis. Both are much lower than the .77 Fourtassi *et al.* (2013) previously documented for English.

Our maximum Token F-score for TP was .66, below the .85 recorded for English and even the .75 recorded for agglutinative languages by Saksida *et al.* (2017). Finally, the score of .4 for DiBS is similar to those recorded for Russian and much lower than the score registered for English (Daland, 2009).

How about the comparison between the two agglutinative languages included here ? Our regression documented an advantage for the less complex Japanese over the more complex Chintang. Although

results thus far confirm the prediction that the *degree* of synthesis affects segmentation, several aspects of our results strongly suggest that the answer is not simple. The language effect here is small and not the same for all models. In other words, even though part of the small effect could be attributed to the fact that both languages are agglutinative, results clearly indicate that morphological complexity is not the sole determinant for word segmentation, and invite a consideration within each algorithm instead.

Our strongest predictions pertained AGu, whose results matched our predictions well, with higher performance for Japanese than Chintang when evaluating on words. This language difference was reduced when evaluating on morphemes. Also consistent with the proposal that AGu, and probably lexical algorithms in general, are ideal to recover recombinable units is the observation that performance was overall higher for morpheme-based than word-based evaluation.

Readers may also notice that AGu achieves the highest scores, providing further evidence to previous observations that lexical models tend to outperform sub-lexical ones (Ludusan *et al.*, 2017). Although this is a desirable feature, we point out that the fact that it is more affected by morphological variation may make it implausible as a strategy for infants learning any and all languages.

As we had predicted by virtue of it being a purely phonotactic-based model, DiBS is less affected by language or level differences. Most token F-scores are similar to the morphologically complex Russian (.35), although markedly lower than English (Daland, 2009). However, Token F-scores obtained with DiBS vary markedly for Japanese morphemes, some of them reaching 0.5. The best explanation for these differences probably requires a recourse to phonological differences across the languages (Daland & Zuraw, 2013), which is orthogonal to our key question.

The most complex patterns are found for prelexical TP. Morphological complexity did not have a systematic effect on performance, as predicted, although we did find an interaction between subtypes, levels, and languages that is not simple to explain briefly.

Scores were higher for morphemes than words for all TPa. This fits in observations by Gervain & Erra 2012, who found that absolute-threshold TP tends to oversegment. When evaluated in morphemes, the greater number of boundaries is not a problem. Contrastingly, better performance for words than morphemes for relative subtypes is not unexpected given that in this class a boundary can only be posited in relatively long strings of syllables, and thus it will tend to undersegment when evaluated in terms of morphemes. TPa may meet both desiderata of high performance and cross-linguistic validity.

5.1 Limitations and future directions

Clearly, our work barely scratches the surface in terms of segmentation differences and similarities across languages. We would look forward to further research incorporating more languages to investigate the impact of linguistic traits (both morpholexical, as studied here, and phonological, as studied elsewhere Daland & Zuraw 2013; Fourtassi *et al.* 2013).

One obvious roadblock facing the generalization of the approach used here to other morphologically varied languages is the sheer paucity of data, since there are overall few corpora of child-directed speech in typologically diverse languages. In ongoing work (Bernard *et al.*, submitted), we are exploring the stability of results as a function of corpus size, to assess what is the minimum size of corpus which would lead to generalizable results. Those analyses suggest that about 5,000 word tokens may suffice – but that analysis was only based on English, and thus further methodological

research is needed to confirm and extrapolate to typologically diverse languages.

But if that approximation were confirmed, then the next roadblock is whether the data that have been (a) morphologically parsed and (b) rendered comparable by e.g. using similar definitions of what a sentence, a word, and a morpheme are. This, we believe, will be an even more challenging obstacle. One of the reasons why we focused on only two languages was because of the way in which they had been carefully curated to be as comparable as possible. Our approach could be generalized to other corpora in AcqDiv which are large enough, although we believe substantial effort would be necessary to generalize it even further to corpora repositories such as CHILDES (MacWhinney, 2014), where, despite clear guidelines seeking to standardize input format, morphological and sentence parsing are ultimately left to the discretion of each corpus' curator.

As additional languages are studied, we hope future researchers retain our strategy of employing a range of plausible models. For instance, as noted above, we focused here on general patterns that were reproduced across different models. However, it would be interesting to "tweak" the models in various ways. One obvious line for exploration would involve changing the input from syllables to phonemes or vice versa, since each of the models used either one or the other, and current results in infant research suggest infants have access to both levels of representation (e.g., Bertoncini & Mehler 1981; Seidl *et al.* 2009).

A more interesting path would be to change the concentration parameters a and b in the adaptor grammar, which, as explained above, govern the reuse versus generation of new lexical items. It is likely that these parameters affect performance, and some previous cross-linguistic work has indeed varied them (Phillips & Pearl, 2014). Similarly, one could build more complex grammars, with various levels of collocation to model the fact that words/morphemes are not independent of previous words – and more generally, that there could be types of words/morphemes typically following each other (e.g., "the" will be followed by a noun or a noun phrase but rarely a verb in English). One consideration this line of research will face is how the child plausibly "decides" which parameters to use and which sets of rules to start with. That is, all of the implementations of the TP family are entirely unsupervised, and DiBS only relies on one supervised parameter which could be replaced in the future with some learning process. Both sublexical families are also extremely simple in terms of their processes and internal architecture. This contrasts even with the AGu system we used, and thus it remains for future work to assess to what extent the whole architecture can be derived in an unsupervised fashion.

5.2 Final conclusions

Both languages studied here yielded lower segmentation scores than those reported in previous work applying the same algorithms to a morphologically simpler language (English). Moreover, a regression suggested lower performance for the more complex of our two languages. However, this regression also suggested complex patterns of interaction depending on the specific model and the level evaluated (i.e., whether word or morpheme boundaries were considered). Future work on additional languages and models would be desirable.

Références

BARTLETT S., KONDRAK G. & CHERRY C. (2009). On the syllabification of phonemes. In *Proceedings of human language technologies : The 2009 annual conference of the north american chapter of the association for computational linguistics*, p. 308–316 : Association for Computational Linguistics.

BATCHELDER E. O. (2002). Bootstrapping the lexicon : A computational model of infant speech segmentation. *Cognition*, **83**(2), 167–206.

BERNARD M., THIOLLIERE R., SAKSIDA A., LOUKATOU G., LARSEN E., JOHNSON M., FIBLA L., DUPOUX E., DALAND R. & CRISTIA X. N. C. A. (submitted). Wordseg : Standardizing unsupervised word form segmentation from text.

BERTONCINI J. & MEHLER J. (1981). Syllables as units in infant speech perception. *Infant behavior and development*, **4**, 247–260.

BICKEL B., GRENOBLE L. A., PETERSON D. A. & TIMBERLAKE A. (2013). *Language typology and historical contingency : In honor of Johanna Nichols*, volume 104. John Benjamins Publishing Company.

BORUTA L., PEPERKAMP S., CRABBÉ B. & DUPOUX E. (2011). Testing the robustness of online word segmentation : Effects of linguistic diversity and phonetic variation. In *Proceedings of the 2nd Workshop on Cognitive Modeling and Computational Linguistics*, p. 1–9 : Association for Computational Linguistics.

CHRISTIANSEN M. H. & CURTIN S. (2005). Integrating multiple cues in language acquisition : A computational study of early infant speech segmentation. *Connectionist models in cognitive psychology*, p. 347–372.

DALAND R. (2009). *Word segmentation, word recognition, and word learning : A computational model of first language acquisition*. PhD thesis, Northwestern University.

DALAND R. & PIERREHUMBERT J. B. (2011). Learning diphone-based segmentation. *Cognitive science*, **35**(1), 119–155.

DALAND R. & ZURAW K. (2013). Does korean defeat phonotactic word segmentation ? In *ACL (2)*, p. 873–877.

DEKEYSER R. M. (2005). What makes learning second-language grammar difficult ? a review of issues. *Language learning*, **55**(S1), 1–25.

FOURTASSI A., BÖRSCHINGER B., JOHNSON M. & DUPOUX E. (2013). Whyisenglishsoeasytosegment. In *Proceedings of the Fourth Annual Workshop on Cognitive Modeling and Computational Linguistics*, p. 1–10.

GERVAIN J. & ERRA R. G. (2012). The statistical signature of morphosyntax : A study of hungarian and italian infant-directed speech. *Cognition*, **125**(2), 263–287.

GOLDWATER S., GRIFFITHS T. L. & JOHNSON M. (2009). A bayesian framework for word segmentation : Exploring the effects of context. *Cognition*, **112**(1), 21–54.

HOFF E. (2013). *Language development*. Cengage Learning.

JAROSZ G. & JOHNSON J. A. (2013). The richness of distributional cues to word boundaries in speech to young children. *Language Learning and Development*, **9**(2), 175–210.

JOHNSON M. (2008). Unsupervised word segmentation for sesotho using adaptor grammars. In *Proceedings of the Tenth Meeting of ACL Special Interest Group on Computational Morphology and gy*, p. 20–27 : Association for Computational Linguistics.

JOHNSON M. & DEMUTH K. (2010). Unsupervised phonemic chinese word segmentation using adaptor grammars. In *Proceedings of the 23rd international conference on computational linguistics*, p. 528–536 : Association for Computational Linguistics.

JOHNSON M., GRIFFITHS T. L. & GOLDWATER S. (2007). Adaptor grammars : A framework for specifying compositional nonparametric bayesian models. In *Advances in neural information processing systems*, p. 641–648.

KUNO S. (1973). *The structure of the Japanese language*, volume 3. MIT press Cambridge, MA.

LIGNOS C. (2011). Modeling infant word segmentation. In *Proceedings of the fifteenth conference on computational natural language learning*, p. 29–38 : Association for Computational Linguistics.

LUDUSAN B., MAZUKA R., BERNARD M., CRISTIA A. & DUPOUX E. (2017). The role of prosody and speech register in word segmentation : A computational modelling perspective. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2 : Short Papers)*, volume 2, p. 178–183.

MACWHINNEY B. (2014). *The CHILDES project : Tools for analyzing talk, Volume II : The database*. Psychology Press.

MCCARTHY J. J. & PRINCE A. S. (1986). *Prosodic morphology*. Wiley Online Library.

MERSAD K. & NAZZI T. (2012). When mommy comes to the rescue of statistics : Infants combine top-down and bottom-up cues to segment speech. *Language Learning and Development*, **8**(3), 303–315.

MONAGHAN P. & CHRISTIANSEN M. H. (2010). Words in puddles of sound : Modelling psycholinguistic effects in speech segmentation. *Journal of child language*, **37**(3), 545–564.

NGON C., MARTIN A., DUPOUX E., CABROL D., DUTAT M. & PEPERKAMP S. (2013). (non) words,(non) words,(non) words : evidence for a protolexicon during the first year of life. *Developmental Science*, **16**(1), 24–34.

PAUDYAL N. P. (2015). *Aspects of Chintang syntax*. PhD thesis, University of Zurich, Philosophische Fakultat.

PHILLIPS L. & PEARL L. (2014). Bayesian inference as a viable cross-linguistic word segmentation strategy : It's all about what's useful. In *Proceedings of the Cognitive Science Society*, volume 36.

PHILLIPS L. & PEARL L. (2015). The utility of cognitive plausibility in language acquisition modeling : Evidence from word segmentation. *Cognitive science*, **39**(8), 1824–1854.

SAFFRAN J. R., ASLIN R. N. & NEWPORT E. L. (1996a). Statistical learning by 8-month-old infants. *Science*, p. 1926–1928.

SAFFRAN J. R., NEWPORT E. L. & ASLIN R. N. (1996b). Word segmentation : The role of distributional cues. *Journal of memory and language*, **35**(4), 606–621.

SAKSIDA A., LANGUS A. & NESPOR M. (2017). Co-occurrence statistics as a language-dependent cue for speech segmentation. *Developmental science*, **20**(3).

SCHIKOWSKI R., PAUDYAL N. & BICKEL B. (2015). Flexible valency in chintang. *Valency Classes : a Comparative Handbook*.

SEIDL A., CRISTIÀ A., BERNARD A. & ONISHI K. H. (2009). Allophonic and phonemic contrasts in infants' learning of sound patterns. *Language Learning and Development*, **5**(3), 191–202.

STOLL S., MAZARA J. & BICKEL B. (2016). The acquisition of polysynthetic verb forms in chintang.

TSUJIMURA N. (2013). *An introduction to Japanese linguistics*. John Wiley & Sons.

VENKATARAMAN A. (2001). A statistical model for word discovery in transcribed speech. *Computational Linguistics*, **27**(3), 351–372.

Évaluation morphologique pour la traduction automatique: adaptation au français

Franck Burlot François Yvon
LIMSI, CNRS, Univ. Paris-Sud, Université Paris Saclay, 91 403 Orsay, France
prénom.nom@limsi.fr

RÉSUMÉ
Le nouvel état de l'art en traduction automatique (TA) s'appuie sur des méthodes neuronales, qui diffèrent profondément des méthodes utilisées antérieurement. Les métriques automatiques classiques sont mal adaptées pour rendre compte de la nature du saut qualitatif observé. Cet article propose un protocole d'évaluation pour la traduction de l'anglais vers le français spécifiquement focalisé sur la compétence morphologique des systèmes de TA, en étudiant leurs performances sur différents phénomènes grammaticaux.

ABSTRACT
Morphological Evaluation for Machine Translation : Adaptation to French

While the state of the art in machine translation has recently changed, it is regularly acknowledged that automatic metrics do not provide enough insights to fully measure the observed qualitative leap. This paper proposes an evaluation protocol for translation from English into French specifically focused on the morphological competence of a system with respect to various grammatical phenomena.

MOTS-CLÉS : Traduction automatique, évaluation de la TA, morphologie.

KEYWORDS: Machine translation, MT evaluation, morphology.

1 Introduction

Le domaine de la traduction automatique (TA) statistique a été récemment transformé par l'arrivée à maturité de nouveaux systèmes de TA reposant massivement sur des architectures neuronales (Sutskever *et al.*, 2014; Bahdanau *et al.*, 2014), qui constituent aujourd'hui le nouvel état de l'art du domaine. Ces nouvelles architectures semblent en particulier capables de détecter (dans la langue source) et de modéliser (dans la langue cible) des dépendances entre mots distants et ainsi de mieux restituer des associations lexicales (collocations, expressions figées) ainsi que des accords grammaticaux (Bentivogli *et al.*, 2016; Isabelle *et al.*, 2017; Sennrich, 2017). Cette amélioration des performances s'est faite au détriment de la prédictibilité et de la transparence des architectures de calcul, dont le fonctionnement s'avère particulièrement opaque et complexe à diagnostiquer.

L'avènement des systèmes neuronaux doit donc s'accompagner du développement de nouvelles méthodes d'évaluation automatique : d'une part parce que le score BLEU (Papineni *et al.*, 2002) ne suffit plus à distinguer des systèmes qui produisent tous des sorties extrêmement fluides ; d'autre part afin de mieux comprendre la capacité des méthodes neuronale à résoudre plus ou moins bien telle ou telle difficulté de traduction, et ainsi d'orienter les évolutions de ces systèmes. Cette ambition a

donné lieu dans les années récentes à une floraison de travaux sur l'évaluation de la TA neuronale, que nous survolons ci-dessous (§ 2.1).

La principale contribution de ce travail est d'étendre l'approche récemment proposée par Burlot & Yvon (2017) pour évaluer les performances morphologiques des architectures neuronales au cas de la traduction vers le français. En plus de traiter une langue supplémentaire, nous introduisons également de nouveaux tests pour le français, qui pourront également être utilisés pour d'autres langues. Après avoir rappelé les principes de la méthode (§ 2), nous décrivons les principaux tests utilisés pour le français (§ 3), puis présentons les résultats d'une comparaison des performances morphologiques de plusieurs systèmes de TA (§ 4), qui permettent d'éclairer l'apport des méthodes neuronales par rapport aux systèmes à base de segments (Koehn, 2010; Allauzen & Yvon, 2011) pour la traduction vers le français. Les scripts et données utilisées dans cette étude sont librement disponibles [1].

2 Principes de l'évaluation morphologique

2.1 Motivations et fondements

La littérature récente sur le diagnostic de TA neuronale peut être organisée en quatre grandes familles : la première s'appuie sur des typologies d'erreurs (Vilar *et al.*, 2006; Lommel *et al.*, 2014) pour catégoriser des erreurs dans les sorties des systèmes, et peuvent impliquer soit une analyse manuelle, souvent difficile ; soit une analyse automatisée (Popović & Ney, 2011; Toral Ruiz & Sánchez-Cartagena, 2017; Klubička *et al.*, 2017), qui se fonde alors sur une comparaison de surface entre la sortie et une traduction de référence [2]. La seconde reprend la tradition ancienne (King & Falkedal, 1990) des jeux de tests (*test suites*) spécifiquement conçus pour mettre en défaut la résolution, par les systèmes de TA, d'un problème linguistique particulier. Isabelle *et al.* (2017) propose un tel jeu de test pour la paire (anglais, français) qui inclut à la fois des difficultés d'ordre morpho-syntaxique (phénomènes d'accord, concordance des modes et temps, etc.), lexical (mots polysémiques, idiomes et expressions figées, etc.), et syntaxique (divergences dans la construction de groupes verbaux, dans la construction de propositions relatives, etc). Cette approche peut être critiquée au regard de l'expertise humaine nécessaire à la création des phrases test comme à l'évaluation des erreurs du système ; par ailleurs, la représentativité des tests et la gravité des erreurs n'est également pas prise en compte.

Une troisième manière de procéder est plus indirecte et consiste à n'utiliser que les scores des systèmes (et non leur sortie) : la qualité d'un système se mesure alors à sa capacité à donner un meilleur score (une plus forte probabilité) à une phrase correcte par rapport à une phrase délibérément altérée pour simuler une faute particulière. Cette méthodologie est utilisée, par exemple, pour évaluer les modèles de langue neuronaux par Linzen *et al.* (2016), qui s'intéressent spécifiquement aux erreurs d'accord (entre sujet et verbe) : le système sera alors considéré comme défaillant s'il assigne à la phrase altérée un meilleur score qu'à la phrase correcte. Sennrich (2017) applique cette stratégie à grande échelle à la TA (de l'anglais vers l'allemand), en engendrant automatiquement des erreurs reflétant des fautes d'accord, des mots inconnus, etc. Si cette méthode permet de s'affranchir de l'intervention d'un expert humain, elle ne permet qu'une évaluation approximative des performances, puisqu'il n'est pas assuré que les deux phrases comparées correspondent à des sorties réelles du système.

1. Voir `https://morpheval.limsi.fr/`

2. Une erreur de morphologie correspond alors à l'observation dans la sortie de la TA d'un lexème présent dans la référence, mais avec une marque flexionnelle différente.

	source	cible
base	he is very happy	il est très heureux
variante	he **was** very happy	il **était** très heureux

FIGURE 1 – Un exemple de test contrastif - après manipulation du temps verbal, on vérifie que la traduction de la variante présente bien un passage au passé par rapport à la traduction de la base.

La méthode proposée par Belinkov *et al.* (2017) est encore plus détournée : elle consiste à comparer les plongements lexicaux (*embeddings*) appris par l'encodeur (ou le décodeur) du système de traduction du point de vue de leur capacité à prédire correctement des tâches de nature morphologique, en l'espèce un étiquetage morpho-syntaxique. Elle n'apporte donc que peu d'information pouvant aider au diagnostic. Si cette approche permet de comparer de manière automatique plusieurs manières de décomposer les mots sources ou cibles, ou de comparer plusieurs couches du réseau, elle dit en revanche peu de chose sur les erreurs morphologiques dans un contexte de traduction (voir également dans la même lignée le travail de Vania & Lopez (2017)).

Comme détaillé ci-dessous, l'approche proposée dans (Burlot & Yvon, 2017) se distingue des méthodes existantes sous divers aspects : (a) elle vise à obtenir un diagnostic *entièrement automatique* portant sur (b) des difficultés morphologiques spécifiques et (c) avérées dans les sorties de traduction automatique ; l'intervention humaine est limitée à la conception des tests, et permet (d) de produire des tests en grande quantité, permettant d'éviter que les systèmes s'adaptent à un jeu de test particulier. La contrepartie est le caractère approximatif de la détection d'erreurs, qui peut toutefois être contrôlé en augmentant le nombre de cas tests.

2.2 Les tests contrastifs

La méthode que nous avions initialement proposée (Burlot & Yvon, 2017) repose sur la notion de *test contrastif*. Dans sa version la plus élémentaire, elle consiste à construire des paires formées de phrases sources comportant une différence minimale : l'une (*la base*) par exemple contiendra un pronom objet masculin, et l'autre (*la variante*) le même objet au féminin. On observe alors les différences entre les traductions de ces deux phrases - lorsqu'elles ne se distinguent que par l'expression (en cible) du trait morphologique qui est manipulé dans la source, on considère que le système a bien reproduit le contraste ; dans le cas inverse on le jugera défaillant.

Deux ensembles de tels tests sont considérés (les ensembles A et B de Burlot & Yvon (2017)). Il existe une seconde famille de tests contrastifs (l'ensemble C) qui se focalisent sur la cohérence des traductions : elle vise à vérifier que les choix de traduction restent cohérents lorsque l'on produit plusieurs variantes de la même base. Ainsi, on pourra remplacer un nom par des synonymes, et vérifier que les propriétés morphologiques de la traduction (en langue cible) de toutes les variantes sont les mêmes et ne dépendent pas de spécificités lexicales : on évalue donc ici plutôt le caractère systématique du fonctionnement du système.

Dans cette approche, les traitements automatiques interviennent à deux étapes : (a) lors de la génération de la ou des variantes, qui demande une analyse grammaticale de la phrase source ; (b) lors du calcul des différences minimales, qui n'exploite que des dictionnaires. Ces deux étapes étant susceptibles d'introduire des erreurs, nous avions proposé de multiplier les paires minimales afin d'obtenir une mesure approximativement correcte du comportement du système. Une évaluation de précision des

Base/Variante(s)	Sortie	Évaluation
	Test-A	
I am hungry	j'ai faim	verbe au passé
I was hungry	j'**avais** faim	trouvé
	Test-B	
I see them	je les vois	le nom et l'adjectif
I see crazy researchers	je vois des **chercheurs fous**	sont au pluriel
	Test-C	
a big responsibility	une **grande** responsabilité	tous les adjectifs
a small responsibility	une **petite** responsabilité	sont au féminin
an important responsibility	une **importante** responsabilité	
a ridiculous responsibility	une responsabilité **ridicule**	
a terrible responsibility	une **terrible** responsabilité	Entropie = 0.0

FIGURE 2 – Exemples de phrases réussissant les tests.

résultats est présentée à la section 4.4.

3 Génération de tests pour le français

3.1 Sélection des phrases tests

La sélection des phrases de base se fonde principalement sur un critère de simplicité, qui accélère la traduction et facilite les traitements automatiques : on se limite à des phrases d'au plus 15 tokens. Ces phrases sont extraites des corpus anglais monolingues News Crawl 2007 et 2008. [3] La production des variantes est plus complexe, en particulier pour les variations lexicales. Le principe général est de remplacer un mot de la base par un mot ou groupe de mots pour produire la variante. Nous utilisons à cet effet l'étiqueteur morpho-syntaxique CoreNLP (Manning *et al.*, 2014) afin de localiser la partie du discours du mot à remplacer, puis le générateur morphologique Pymorphy [4] pour produire la flexion désirée. La génération de synonymes et d'antonymes est effectuée avec WordNet (Miller, 1995). Une dernière étape consiste à employer un modèle de langue (Heafield, 2011) entraîné sur toutes les données monolingues anglaises mises à disposition pour la campagne WMT 2015 [5] pour sélectionner les variantes les plus fluides. Cette sélection aboutit à 500 groupes de phrases pour chaque test.

3.2 Test-A : transformations morpho-syntaxiques

Le premier groupe de tests consiste à modifier la flexion d'un mot dans la base, puis à évaluer la présence du même contraste côté cible. Les tests présentés ici adaptent, pour partie, les propositions de Burlot & Yvon (2017). C'est le cas des tests portant (a) sur le temps des verbes, où un verbe au présent est remplacé respectivement par les formes au `passé` et au `futur` ; (b) sur le `nombre` d'un

3. `http://statmt.org/wmt17/translation-task.html`
4. `http://pymorphy.readthedocs.io/`
5. `http://statmt.org/wmt15/translation-task.html`

pronom objet initialement au singulier ; (c) sur la `négation` d'une base à l'affirmatif ; (d) sur le `comparatif` où un adjectif au comparatif dans la base est remplacé par une forme neutre.

Le français présentant une morphologie verbale riche, deux tests supplémentaires sont proposés. Le premier évalue la génération du `conditionnel` : pour ce faire, nous remplaçons l'auxiliaire *will* dans la base par le modal *would* et testons le contraste indicatif/conditionnel dans les phrases cibles. Le second évalue le passage d'un verbe de l'indicatif au `subjonctif` ; les paires contrastives sont produites en recherchant des phrases introduites par des propositions principales du type *I believe*, changées dans les variantes en *I don't believe*. Il est attendu que, pour la variante, la traduction de la proposition subordonnée comprenne un verbe au subjonctif. Enfin, nous ajoutons également un test concernant le `superlatif`, produit de manière identique à celui du comparatif.

De manière générale, l'évaluation se déroule de la manière suivante : chacun des mots de la variante traduite qui n'est pas présent dans la base traduite est récolté et analysé au moyen du dictionnaire Lefff[6] (Sagot, 2010). Si le dictionnaire ne propose aucune analyse pour le mot trouvé, la paire de phrase est rejetée du test, puisqu'il est dans ce cas impossible de déterminer si elle reflète ou non le contraste attendu. Ne sont retenus que les mots qui contiennent dans leur analyse la partie du discours évaluée (par exemple les verbes dans le test du subjonctif). Enfin, l'étiquette morphologique reflétant le phénomène grammatical du test est recherchée : si elle est présente, un succès est rapporté.

3.3 Test-B : transformations lexicales

Ces tests ont pour objectif d'évaluer la capacité d'un système à modéliser différentes formes d'accord grammatical. De même que dans la partie précédente, certains tests sont repris de Burlot & Yvon (2017). Ainsi, `verbes coord` consiste à changer le verbe de la base en un groupe verbal coordonné (*he eats* → *he eats and drinks*), puis à vérifier dans la traduction que les deux verbes coordonnés contiennent bien les mêmes marques de nombre, de personne et temps/mode. Le test de syntagmes nominaux (`synt nom`) est similaire : un pronom dans la base est modifié en syntagme nominal `ADJ+NOM` dans la variante. L'évaluation vérifie (séparément) l'accord en genre et l'accord en nombre entre l'adjectif et le nom dans la cible française.

Un nouveau test (`coréf`) concerne les liens de coréférence qui existent entre un pronom personnel et son antécédent nominal. Les bases sont sélectionnées lorsqu'elles contiennent un lien de coréférence détecté par l'étiqueteur de CoreNLP[7] (Manning *et al.*, 2014) ; l'antécédent nominal dans la base est substitué par un synonyme et l'on vérifie alors que les pronoms sont correctement accordés en nombre ou en genre.

Notons que pour `verbes coord` et `coréf`, l'utilisation de contrastes permet de projeter des annotations depuis l'anglais vers le français, ce qui est un usage quelque peu différent de celui utilisé pour le premier jeu de tests. Pour `coréf`, en observant ce qui co-varie dans les traductions de la base et de la variante, il devient possible de localiser l'antécédent du pronom dans les deux phrases cibles. Dans ce cas, chaque paire nous permet d'évaluer deux traductions, et les scores sont récoltés sur la base et sur la variante.

6. http ://alpage.inria.fr/ sagot/lefff.html

7. Afin de privilégier la précision aux dépens du rappel, nous avons conservé les bases étiquetées positivement à la fois par le modèle de base et par le modèle neuronal de la boîte à outils.

	verbes							
Système	passé		futur		cond.		subj.	
Moses	69,5%	321/462	88,7%	422/476	62,3%	299/480	86,0%	430/500
Nematus	74,4%	349/469	76,2%	356/467	62,9%	303/482	85,5%	425/497
+rétro-trad.	84,3%	402/477	85,2%	410/481	83,1%	398/479	91,4%	457/500

	pronoms		adjectifs				autres			
Système	nb		compar.		super.		nég.		noms plur.	
Moses	82,8%	414/500	80,4%	402/500	78,0%	390/500	97,0%	485/500	82,6%	395/478
Nematus	76,4%	382/500	82,2%	411/500	87,4%	437/500	96,6%	483/500	86,8%	400/461
+rétro-tr.	87,8%	439/500	87,2%	436/500	90,2%	451/500	98,8%	494/500	89,2%	415/465

TABLE 1 – Évaluation des paires de phrases (test-A).

	verbes coord.					
Système	nb.		pers.		TM	
Moses	97,5%	394/404	97,0%	392/404	95,3%	385/404
Nematus	94,8%	423/446	94,6%	422/446	94,4%	421/446
+rétro-trad.	97,8%	435/445	98,4%	438/445	98,0%	436/445

	synt. nom.				coréférence	
Système	genre		nb.		genre	
Moses	94,4%	356/377	92,0%	347/377	83,2%	691/831
Nematus	95,8%	365/381	95,8%	365/381	89,4%	787/880
+rétro-trad.	97,9%	375/383	98,4%	377/383	88,4%	827/936

TABLE 2 – Évaluation des paires de phrases (test-B).

3.4 Test-C : tests de cohérence

La troisième famille de tests est quelque peu différente : pour chaque base, on produit 4 variantes, et l'on mesure la cohérence des choix du système de TA en mesurant l'entropie du trait morphologique contrôlé : un système idéal doit produire toujours les mêmes traits (ce qui correspond à une entropie nulle), alors qu'un système incohérent produira une entropie maximale. Cinq tests sont considérés, trois qui portent sur les verbes (respectivement sur le nombre, le genre, et le couple (temps, mode) TM , et deux sur les adjectifs (pour le genre et le nombre). Pour chacun, on rapporte l'entropie moyennée sur tous les groupes de phrases du test.

Système	BLEU
Moses	32,25
Nematus	33,06
+rétro-trad.	34,11

TABLE 3 – Scores BLEU (Newstest-2014).

	verbes			adjectifs	
Système	nb.	pers.	TM	genre	nb.
Moses	0,075	0,039	0,099	0,131	0,131
Nematus	0,040	0,033	0,080	0,076	0,052
+rétro-trad.	0,024	0,015	0,066	0,065	0,049

TABLE 4 – Évaluation des groupes de phrases (test-C).

4 Évaluation

4.1 Systèmes et données

Les systèmes choisis pour illustrer notre méthode d'évaluation de la morphologie sont représentatifs de l'évolution récente de la TA et nous en présentons ici deux types : statistique et neuronal.

Le système statistique est basé sur Moses (Koehn *et al.*, 2007). Il est entraîné sur 4 millions de phrases parallèles provenant des données fournies à WMT 2017 (plus précisément des corpus EUbookshop, MultiUN, News-Commentary-11 et Wikipedia). Le modèle de langue employé par le système a été entraîné avec KenLM (Heafield, 2011) sur le côté cible des données parallèles, auxquelles ont été ajoutées environ 10 millions de phrases issues du corpus news-2014 fourni à WMT 2015.

Le système neuronal a été entraîné avec la boîte à outils Nematus (Sennrich *et al.*, 2017) sur les mêmes phrases parallèles que le système statistique. Les vocabulaires source et cible ont été traités avec un modèle bilingue de *Byte Pair Encoding* (Sennrich *et al.*, 2016b), paramétré à 50 000 opérations de fusion. Ce traitement a conduit à un vocabulaire anglais de plus de 32 000 unités et à un vocabulaire français de moins de 36 000 unités.

Un second traducteur neuronal reprend les paramètres du système précédent, qui sont employés pour initialiser l'entraînement d'un nouveau système optimisé sur 2 millions de phrases sélectionnées aléatoirement parmi les données initiales, complété par 2 millions de phrases françaises extraites du corpus Europarl (Koehn, 2005) *rétro-traduites* (Sennrich *et al.*, 2016a) vers l'anglais au moyen d'un système neuronal français-anglais similaire au système décrit supra pour la direction anglais-français.

4.2 Résultats

Les scores BLEU (Papineni *et al.*, 2002) pour ces trois systèmes ont été calculés sur Newstest-2014 et sont dans le tableau 3. Ils distinguent sensiblement le système statistique des systèmes neuronaux qui obtiennent 1 à 2 points de plus. Les précisions obtenues sur les tâches du test-A apparaissent au tableau 1 et contredisent quelque peu les scores BLEU. C'est ce que l'on observe pour le futur, qui obtient la plus haute précision, mais aussi pour le conditionnel et le subjonctif, dont les précisions sont similaires au système Nematus. Cela révèle l'efficacité relative des systèmes neuronaux dans la transmission d'une caractéristique morphologique de la source vers la cible. En effet, si ces systèmes sont réputés pour fournir une sortie plus fluide que les modèles statistiques, cela se produit parfois aux dépens de l'adéquation de la cible avec la source.

Enfin, les comparatifs et superlatifs semblent mieux pris en charge par les systèmes neuronaux. Nous posons l'hypothèse que cela est dû au caractère ouvert de leurs vocabulaires, qui permettent ainsi de générer potentiellement n'importe quel mot forme. À l'opposé, les systèmes statistiques ont un vocabulaire fixe et lorsqu'un adjectif anglais au comparatif ou au superlatif n'a pas été observé à l'entraînement, le système est incapable de générer une traduction correcte. Cette remarque tend à expliquer la meilleure performance du système statistique par rapport au système neuronal sans rétro-traduction sur la tâche des pronoms au pluriel. En effet, les pronoms correspondant à une classe de mots fermée, Moses n'a aucune difficulté à les traiter correctement. Lorsqu'en revanche ce système est confronté à une classe ouverte, comme celle des noms au pluriel, le caractère fixe de son vocabulaire limite ses performances.

tâche	Fréquences (f) des mots produits dans la variante							
	$f = 0$		$0 < f < 50$		$50 < f < 1000$		$1000 < f$	
passé	0,0%	(0/0)	81,2%	(13/16)	71,7%	(66/92)	68,5%	(241/352)
comparatif	0,0%	(0/2)	50,0%	(9/18)	76,9%	(60/78)	79,0%	(226/286)
noms plur.	0,0%	(0/1)	79,5%	(31/39)	79,0%	(94/119)	84,9%	(270/318)
v. tps/mode	87,5%	(7/8)	97,7%	(42/43)	98,4%	(121/123)	92,3%	(120/130)
coréf. genre	40,0%	(2/5)	54,8%	(34/62)	68,3%	(153/224)	71,1%	(494/695)

TABLE 5 – Performance du système Moses selon la fréquence des mots à traduire.

tâche	Fréquences (f) des mots produits dans la variante							
	$f = 0$		$0 < f < 50$		$50 < f < 1000$		$1000 < f$	
passé	50,0%	(1/2)	61,1%	(11/18)	76,7%	(66/86)	87,3%	(322/369)
comparatif	0,0%	(0/2)	11,1%	(2/18)	82,1%	(64/78)	89,5%	(256/286)
noms plur.	66,7%	(2/3)	68,6%	(24/35)	96,5%	(110/114)	89,4%	(279/312)
v. tps/mode	100,0%	(10/10)	95,1%	(39/41)	98,5%	(133/135)	99,3%	(139/140)
coréf. genre	60,0%	(3/5)	69,4%	(43/62)	81,2%	(182/224)	84,5%	(587/695)

TABLE 6 – Performance du système Nematus + rétro-traduction selon la fréquence des mots à traduire.

Les tests-B (tableau 2) ne révèlent pas de différences aussi importantes entre les systèmes. Nous constatons toutefois que le système Nematus sans rétro-traduction est le plus mauvais pour le test de coordination verbale. Notons que le système statistique emploie un modèle de langue entraîné sur une grande quantité de données monolingues, ce qui n'est pas le cas des systèmes neuronaux qui n'ont observé que quelques millions de phrases cibles. Il est donc indéniable que les modèles neuronaux tirent un bien meilleur parti d'une moindre quantité de données monolingues. La tâche de coréférence place enfin le système statistique en-dessous des systèmes neuronaux qui semblent mieux prendre en charge de tels phénomènes d'accord distants.

Les tests-C (tableau 4) témoignent d'une progression claire mettant en valeur la supériorité du système neuronal avec rétro-traduction, qui a toujours une entropie inférieure aux deux autres. La variété lexicale inhérente à ce test révèle la faiblesse du système statistique qui peine à rester cohérent dans sa prédiction morphologique lorsque le syntagme nominal n'a pas été observé à l'entraînement.

4.3 Performance sur les mots rares et inconnus

Nous proposons ici d'affiner les résultats présentés en mesurant la performance d'un système selon la fréquence du mot source sur lequel porte une tâche. Ces fréquences sont calculées sur le côté source des mêmes données parallèles employées par chaque système. Des précisions sont ainsi rapportées pour un sous-ensemble de tâches aux tableaux 5 (statistique) et 6 (neuronal avec rétro-traduction). Ainsi, pour la tâche du `passé`, nous considérons le mot source introduit dans chaque variante qui porte la marque du passé. Les phrases sont classées selon quatre intervalles de fréquences : les mots inconnus, les mots dont la fréquence est inférieure à 50, inférieure à 1000 et supérieure à 1000.

Nous pouvons constater que la génération des variantes lors de la création du jeu de test a produit peu ou pas de mots inconnus. Le système neuronal est basé sur une segmentation des mots en BPE, ce qui lui permet en théorie d'interpréter n'importe quel mot inconnu par la combinaison de plusieurs unités en source. Toutefois, nous n'observons pas ici d'amélioration significative du système neuronal par

tâche	moyenne	1000	750	500	250	100
			Test-A			
passé	76.8	±2.6	±3.0	±3.7	±5.2	±8.2
futur	83.6	±2.3	±2.6	±3.2	±4.6	±7.2
conditionnel	81.9	±2.4	±2.8	±3.4	±4.8	±7.5
subjonctif	92.8	±1.6	±1.8	±2.3	±3.2	±5.0
pron. nb	85.0	±2.2	±2.6	±3.1	±4.4	±6.9
comparatif	80.7	±2.4	±2.8	±3.5	±4.9	±7.7
superlatif	91.4	±1.7	±2.0	±2.5	±3.5	±5.4
négation	97.2	±1.0	±1.2	±1.4	±2.0	±3.0
noms plur.	85.8	±2.2	±2.5	±3.1	±4.3	±6.8
			Test-B			
v. nombre	94.8	±1.4	±1.6	±1.9	±2.7	±4.2
v. personne	94.4	±1.4	±1.6	±2.0	±2.8	±4.4
v. tps/mode	93.9	±1.5	±1.7	±2.1	±2.9	±4.6
SN genre	96.7	±1.1	±1.3	±1.6	±2.2	±3.4
SN nombre	98.1	±0.8	±1.0	±1.2	±1.6	±2.4
coréf. genre	89.7	±1.9	±2.2	±2.7	±3.7	±5.9

TABLE 7 – Significativité des mesures avec le système Nematus + rétro-traduction

rapport au système statistique.

Comme l'on pouvait s'y attendre, plus la fréquence augmente et plus la performance est élevée. Le nombre de phrases comportant des fréquences inférieures à 50 est généralement réduit, toutefois, nous constatons que le système statistique à tendance à mieux gérer ces mots rares : passé, comparatif, noms plur. et v. tps/mode présentent dans ce cas une plus grande précision. En revanche, au-delà de 50 occurrences, strictement toutes les précisions du système neuronal sont supérieures. Ce résultat tend à montrer que le problème des mots rares reste important en traduction neuronale, et leur segmentation en unités plus fréquentes ne garantit pas une bonne traduction.

4.4 Une évaluation de la métrique

4.4.1 Significativité des scores

Chaque tâche introduite jusqu'ici suppose une précision (ou une entropie) calculée sur la base de 500 paires (ou groupes) de phrases. Pour mesurer l'impact de la quantité d'exemples sur les mesures de qualité, nous estimeons la significativité des scores obtenus selon différentes tailles du jeu de tests.

Nous avons constitué un nouveau jeu de test indépendant du premier et comportant 1000 exemples par tâche, qui a été traduit avec le système neuronal bénéficiant des données rétro-traduites (tableau 7). Sont considérés différents sous-ensembles comprenant 100 à 1000 exemples issus des tests A et B. Pour chacun d'entre eux, 10 000 différents tests de la même taille ont été échantillonnés aléatoirement parmi les 1000 disponibles. Des scores moyens ont ainsi été obtenus, ainsi qu'une mesure de significativité, selon un intervalle de confiance de 95%.

Ces mesures de significativité ont été réalisées avec pour objectif de rechercher un nombre d'exemples par tâche qui satisfasse deux considérations opposées. D'une part, ce nombre doit être assez élevé pour permettre un calcul de précisions assez fin dans le but de comparer deux systèmes. D'autre part, il doit être limité pour que le jeu de test ne soit pas trop volumineux, ce qui conduirait à des temps de

décodage trop longs et peu pratiques pour les systèmes neuronaux.

Nous constatons ainsi que la différence entre les système Moses et Nematus+rétro-traduction sur la tâche du `comparatif` (respectivement 80,4 et 87,2) n'aurait pas été significative si elle avait été obtenue sur 100 exemples ($\pm$ 7,7). Sur 500 exemples, et avec variation de $\pm$ 3,5, nous sommes en mesure de distinguer les deux systèmes avec un certain degré de confiance. Nous constatons par ailleurs que tous deux ne montrent pas de différence significative ($\pm$ 1,4) avec 500 exemples pour la tâche de `v. nombre` (respectivement 97,5 et 97,8). Notons qu'une telle différence n'aurait pas été plus pertinente sur 1000 exemples ($\pm$ 1,4). Ainsi, notre choix de sélectionner 500 exemples permet d'avoir des tests significatifs et relativement rapides à mettre en œuvre.

4.4.2 Évaluation qualitative

Nous présentons ici quelques exemples concrets de traduction et de leur évaluation, telle qu'elle s'opère automatiquement dans le cadre du protocole présenté.

La tâche `SN nombre` consiste à changer dans la source un pronom en syntagme nominal et de vérifier si un système modélise correctement l'accord entre l'adjectif et le nom.

source	I don't want to kill **you**. I don't want to kill the **impartial compositors**.	
Moses	Je ne veux pas vous tuer. Je ne veux pas tuer les **clavistes impartial**.	✗
Nematus	je ne veux pas tuer . Je ne veux pas tuer les **compositeurs impartiaux**.	✓
+ rétro-trad.	Je ne veux pas vous tuer. Je ne veux pas tuer les **compositeurs impartiaux**.	✓

Le mot *compositors* est peu fréquent dans les données d'entraînement, si bien que le modèle de langue du système Moses n'a observé ni *clavistes impartial*, ni *clavistes impartiaux*, bien que les deux formes de l'adjectif soient présentes dans les données. En revanche, le système neuronal bénéficie d'une plus grande généralisation en segmentant l'adjectif en *imparti- aux*, ce qui distingue une terminaison univoque du pluriel.

Il arrive toutefois que certaines erreurs syntaxiques du système Moses ne puissent pas être repérées dans notre protocole. C'est ce que l'on observe dans cet exemple de la tâche `passé`, où la variante générée est une mauvaise traduction, mais où le test est néanmoins réussi, puisqu'un verbe au passé a été automatiquement détecté.

source	That **prompts** Tara to ask when she can eat. That **prompted** Tara to ask when she can eat.	
Moses	Tara qui se demander quand elle peut manger. Tara, qui **ont conduit** à se demander si elle peut manger.	✓
Nematus	Cela amène Tara à se demander quand elle peut manger. Cela **a incité** Tara à se demander quand elle peut manger.	✓
+ rétro-trad.	Cela incite Tara à se demander quand elle peut manger. Cela **a poussé** Tara à se demander quand elle peut manger.	✓

Dans le cas où la tâche `SN nombre` présente un mot inconnu dans les données,(ici *signallers*), notre protocole permet d'écarter facilement l'hypothèse fournie par Moses qui ne fait qu'une copie. Dans

ces cas, les systèmes neuronaux sont capables de générer des phrases parfaitement fluides, quoique peu fidèles à la source. Notre protocole consiste ici à repérer dans la variante deux mots qui sont absents de la base et à vérifier leur accord : ainsi, *signaux truqués* et *messages truqués* reflètent bien l'accord voulu, et nous ne sommes pas en mesure de juger la qualité de la traduction.

source	What more do **you** need to say ? What more do the **truthful signallers** need to say ?	
Moses	Plus ce que vous voulez dire ? Ce qu' il faut faire la **vérité signallers** à dire ?	✗
Nematus	Qu'avez-vous besoin de dire ? Qu'en est-il des **signaux truqués** ?	✓
+ rétro-trad.	Qu'est-ce que vous devez dire ? Quels sont les **messages truqués** qu'il faut dire ?	✓

Des erreurs peuvent par ailleurs provenir de la génération du test. Dans cet exemple de la tâche `futur`, l'analyseur en parties du discours a interprété *call* dans la base comme un verbe. La variante qui en résulte est donc agrammaticale et aucune traduction correcte ne peut être attendue.

source	Telephone **calls** to Khan and Kearney were not immediately returned. Telephone **will call** to Khan and Kearney were not immediately returned.	
Moses	Les appels téléphoniques à Khan et Kearney n'étaient pas immédiatement retourné. Téléphone fera appel à Khan et Kearney n'étaient pas immédiatement retourné.	✓
Nematus	Les appels téléphoniques à Khan et à Kearney n'ont pas été immédiatement retournés. Le téléphone fera appel à Khan et à Kearney.	✓
+ rétro-trad.	Les appels téléphoniques à Khan et à Kearney n'ont pas été immédiatement renvoyés. Téléphoner à Khan et à Kearney n'a pas été immédiatement retourné.	✗

Une fois encore, nous observons la grande fluidité des traductions neuronales, même lorsque les systèmes ne parviennent pas à interpréter la source correctement. En effet, le système statistique se montre incapable de traduire une séquence de deux verbes d'état, ce que la traduction neuronale résout dans la tâche `v. tps/mode`.

source	Our responsibility **lies** in communicating this information ! Our responsibility **rests and always lies** in communicating this information !	
Moses	Notre responsabilité est de communiquer cette information ! Notre responsabilité **est toujours et communiquer** cette information !	✗
Nematus	Notre responsabilité réside dans la communication de cette information ! Notre responsabilité **repose et réside toujours** dans la communication de cette information !	✓
+ rétro-trad.	Notre responsabilité réside dans la communication de cette information ! Notre responsabilité **repose et réside toujours** dans la communication de cette information !	✓

Une tâche semble toutefois rester difficile pour tous les systèmes : la `coréférence`. Ici, le système statistique omet tout simplement les pronoms. Les modèles neuronaux produisent bien un pronom, mais le système +rétro-trad semble choisir *administration* comme antécédent du pronom, ce qui le conduit à prédire le mauvais genre. Quant au système Nematus, il génère un pronom correct pour la base et un pronom ambigu du point de vue de la tâche : *l'* pouvant être à la fois masculin et féminin, il est toujours considéré comme correct.

source	The Bush administration should support the UN **process** and not undermine it. The Bush administration should support the UN **effort** and not undermine it.	
Moses	L'administration Bush doit soutenir le **processus** de l'ONU et ne pas saper. L'administration Bush doit appuyer les **efforts** de l'ONU et ne pas saper.	✗ ✗
Nematus	L'administration Bush devrait soutenir le **processus** de l'ONU et ne pas **le** saper. L'administration Bush devrait soutenir l'**effort** de l'ONU et ne pas **l'**affaiblir.	✓ ✓
+ rétro-trad.	L'administration Bush devrait soutenir le **processus** des Nations unies et ne pas **la** saper. L'administration Bush devrait soutenir l'**effort** des Nations unies et ne pas **la** saper.	✗ ✗

Cette évaluation permet également de distinguer les deux systèmes neuronaux. Dans la tâche `conditionnel`, Nematus ne traduit de la source ni le sens, ni la valeur du conditionnel. Le système +rétro-trad. est lui capable de produire une traduction fidèle à la source, tout en produisant le conditionnel attendu dans la variante. Ce système bénéficie de données synthétiques (rétro-traduites automatiquement) qui sont plus littérales que les traductions humaines (Crego & Senellart, 2016), et permettent au système de mieux transférer un sens grammatical depuis la source.

source	That is what **will** keep you alive. That is what **would** keep you alive.	
Moses	C'est ce que vous **permettront** de maintenir en vie. C'est ce que vous **permettre** de maintenir en vie.	✗
Nematus	C'est ce qui **va** rester en vie. C'est ce qui **est** en vie.	✗
+ rétro-trad.	C'est ce qui vous **tiendra** en vie. C'est ce qui vous **tiendrait** en vie.	✓

5 Conclusion

Dans cet article, nous avons présenté un protocole d'évaluation de la TA depuis l'anglais vers le français spécialisé dans l'analyse de la compétence morphologique des systèmes. Contrairement aux métriques automatiques qui mettent en évidence la supériorité des systèmes neuronaux sur les systèmes statistiques, les tests présentés suggèrent que certains phénomènes grammaticaux sont moins bien modélisés dans la traduction neuronale, notamment lorsqu'il s'agit de transmettre une caractéristique morphologique depuis la source, ou lorsqu'il s'agit de traduire des mots peu fréquents.

Le protocole présenté implique la génération automatique d'un jeu de tests, au cours de laquelle certaines erreurs peuvent apparaître. Pour diminuer l'impact de ces erreurs, il est trivial d'augmenter le nombre d'exemples, dans les limites imposées par le coût de décodage ; 500 exemples par tâche semblant constituer un bon compromis. L'avantage d'une telle approche réside dans son caractère automatique, qui réduit l'intervention humaine à la conception de tâches. L'analyse est donc basée sur de nombreux exemples et permet une focalisation sur des phénomènes linguistiques précis.

Remerciements

Ce travail a été partiellement financé par le programme H2020 de l'Union Européenne dans le cadre de l'accord de subvention No. 645452 (QT21).

Références

ALLAUZEN A. & YVON F. (2011). Méthodes statistiques pour la traduction automatique. In E. GAUSSIER & F. YVON, Eds., *Modèles Probabilistes pour l'accès à l'information*, chapter 7, p. 271–356. Hermès, Paris.

BAHDANAU D., CHO K. & BENGIO Y. (2014). Neural machine translation by jointly learning to align and translate. *CoRR*, **abs/1409.0473**.

BELINKOV Y., DURRANI N., DALVI F., SAJJAD H. & GLASS J. (2017). What do neural machine translation models learn about morphology ? In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1 : Long Papers)*, p. 861–872.

BENTIVOGLI L., BISAZZA A., CETTOLO M. & FEDERICO M. (2016). Neural versus phrase-based machine translation quality : a case study. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, p. 257–267, Austin, Texas.

BURLOT F. & YVON F. (2017). Evaluating the morphological competence of machine translation systems. In *Proceedings of the Second Conference on Machine Translation, Volume 1 : Research Papers*, p. 43–55, Copenhagen, Denmark.

CREGO J. M. & SENELLART J. (2016). Neural machine translation from simplified translations. *CoRR*, **abs/1612.06139**.

HEAFIELD K. (2011). KenLM : Faster and Smaller Language Model Queries. In *Proceedings of the Sixth Workshop on Statistical Machine Translation*, p. 187–197, Edinburgh, Scotland.

ISABELLE P., CHERRY C. & FOSTER G. (2017). A challenge set approach to evaluating machine translation. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, p. 2486–2496, Copenhagen, Denmark.

KING M. & FALKEDAL K. (1990). Using test suites in evaluation of machine translation systems. In *COLNG 1990 Volume 2 : Papers presented to the 13th International Conference on Computational Linguistics*, Helsinki, Finland.

KLUBIČKA F., TORAL RUIZ A. & SÁNCHEZ-CARTAGENA V. M. (2017). Fine-grained human evaluation of neural versus phrase-based machine translation. In *Proceedings of the European Conference on Machine Translation*, EAMT'17, p. 121—-132, Prague, Czech Republic.

KOEHN P. (2005). Europarl : A Parallel Corpus for Statistical Machine Translation. In *Conference Proceedings : the tenth Machine Translation Summit*, p. 79–86, Phuket, Thailand : AAMT AAMT.

KOEHN P. (2010). *Statistical Machine Translation*. Cambridge University Press.

KOEHN P., HOANG H., BIRCH A., CALLISON-BURCH C., FEDERICO M., BERTOLDI N., COWAN B., SHEN W., MORAN C., ZENS R., DYER C., BOJAR O., CONSTANTIN A. & HERBST E. (2007). Moses : Open source toolkit for statistical machine translation. In *Proceedings of the 45th Annual Meeting of the ACL on Interactive Poster and Demonstration Sessions*, ACL '07, p. 177–180, Stroudsburg, PA, USA : Association for Computational Linguistics.

LINZEN T., DUPOUX E. & GOLDBERG Y. (2016). Assessing the ability of LSTMs to learn syntax-sensitive dependencies. *Transactions of the Association for Computational Linguistics*, **4**, 521–535.

LOMMEL A., BURCHARDT A., POPOVIC M., HARRIS K., AVRAMIDIS E. & USZKOREIT H. (2014). Using a new analytic measure for the annotation and analysis of mt errors on real data. In *Proceedings of the conference of the European Association for Machine Translation*, EAMT 2014, Dubrovnik, Croatia.

MANNING C. D., SURDEANU M., BAUER J., FINKEL J., BETHARD S. J. & MCCLOSKY D. (2014). The Stanford CoreNLP natural language processing toolkit. In *Association for Computational Linguistics (ACL) System Demonstrations*, p. 55–60.

MILLER G. A. (1995). WordNet : A Lexical Database for English. *Communications of the ACM*, **38**(11), 39–41.

PAPINENI K., ROUKOS S., WARD T. & ZHU W.-J. (2002). BLEU : a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, p. 311–318, Stroudsburg, PA, USA : Association for Computational Linguistics.

POPOVIĆ M. & NEY H. (2011). Towards automatic error analysis of machine translation output. *Computational Linguistics*, **37**(4), 657–688.

SAGOT B. (2010). The Lefff, a freely available and large-coverage morphological and syntactic lexicon for French. In *7th international conference on Language Resources and Evaluation (LREC 2010)*, Valletta, Malta.

SENNRICH R. (2017). How grammatical is character-level neural machine translation ? assessing mt quality with contrastive translation pairs. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics : Volume 2, Short Papers*, p. 376–382 : Association for Computational Linguistics.

SENNRICH R., FIRAT O., CHO K., BIRCH A., HADDOW B., HITSCHLER J., JUNCZYS-DOWMUNT M., LÄUBLI S., BARONE A. V. M., MOKRY J. & NADEJDE M. (2017). Nematus : a toolkit for neural machine translation. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics, EACL 2017, Valencia, Spain, April 3-7, 2017, Software Demonstrations*, p. 65–68.

SENNRICH R., HADDOW B. & BIRCH A. (2016a). Improving neural machine translation models with monolingual data. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1 : Long Papers)*, p. 86–96, Berlin, Germany.

SENNRICH R., HADDOW B. & BIRCH A. (2016b). Neural machine translation of rare words with subword units. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1 : Long Papers)*, p. 1715–1725.

SUTSKEVER I., VINYALS O. & LE Q. V. (2014). Sequence to sequence learning with neural networks. In *Proceedings of the 27th International Conference on Neural Information Processing Systems - Volume 2*, NIPS'14, p. 3104–3112, Cambridge, MA, USA : MIT Press.

TORAL RUIZ A. & SÁNCHEZ-CARTAGENA M. (2017). A multifaceted evaluation of neural versus phrase-based machine translation for 9 language directions. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics : Volume 1, Long Papers*, p. 1063–1073, Valencia, Spain : Association for Computational Linguistics (ACL).

VANIA C. & LOPEZ A. (2017). From characters to words to in between : Do we capture morphology ? In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1 : Long Papers)*, p. 2016–2027 : Association for Computational Linguistics.

VILAR D., XU J., LUIS FERNANDO D. & NEY H. (2006). Error analysis of statistical machine translation output. In *Proceedings of the Fifth International Conference on Language Resources and Evaluation*, LREC'06, Genoa, Italy.

Étiquetage en parties du discours de langues peu dotées par spécialisation des plongements lexicaux

Pierre Magistry[1] Anne-Laure Ligozat[2] Sophie Rosset[1]
(1) LIMSI, CNRS, Université Paris-Saclay, Bât 508, rue John von Neumann, Campus Universitaire, F-91405 Orsay
(2) LIMSI, CNRS, ENSIIE, Université Paris-Saclay, Bât 508, rue John von Neumann, Campus Universitaire, F-91405 Orsay
{magistry,annlor,rosset}@limsi.fr

RÉSUMÉ
Cet article présente une nouvelle méthode d'étiquetage en parties du discours adaptée aux langues peu dotées : la définition du contexte utilisé pour construire les plongements lexicaux est adaptée à la tâche, et de nouveaux vecteurs sont créés pour les mots inconnus. Les expériences menées sur le picard, le malgache et l'alsacien montrent que cette méthode améliore l'état de l'art pour ces trois langues peu dotées.

ABSTRACT
POS tagging for low-resource languages by adapting word embeddings

This paper presents a new method for Part-of-speech tagging, adapted to low-resource languages : the context definition is adapted to the POS tagging task, and new vectors are created for unknown words. Experiments on Picard, Malagasy and Alsatian show that it improves the state of the art for all three languages.

MOTS-CLÉS : étiquetage en parties du discours, langues peu dotées.

KEYWORDS: POS tagging, low resource languages.

1 Introduction

De très grands corpus bruts et annotés sont désormais disponibles pour certaines langues, et les méthodes par apprentissage en Traitement automatique des langues s'appuient maintenant souvent sur leur existence : les corpus bruts permettent d'apprendre des plongements lexicaux, et les corpus annotés servent à l'apprentissage des modèles. Cependant, l'écrasante majorité des langues du monde ne dispose pas de telles ressources, et les méthodes existantes ne peuvent donc pas être utilisées sans adaptation.

En ce qui concerne l'étiquetage en parties du discours, le manque de ressources pour les langues peu dotées pose deux problèmes : la faible quantité de corpus bruts ne permet pas d'apprendre des plongements lexicaux de même qualité que pour les langues généralement étudiées ; par ailleurs, les données annotées sont également rares, et l'annotation de nouveaux corpus peut être difficile, notamment en raison de la difficulté à trouver des annotateurs. Plusieurs stratégies sont possibles pour pallier ces problèmes, par exemple s'appuyer sur des corpus parallèles ou sur des ressources de

langues proches. Cependant, nous proposons une autre stratégie, qui nous permet de nous situer dans le cadre le plus général possible : pas de corpus parallèle, pas de langue proche bien dotée.

Dans cet article, nous proposons donc une nouvelle méthode pour l'étiquetage en parties du discours de langues peu dotées, qui s'appuie sur une méthode de construction adaptée des plongements lexicaux. Cette méthode est indépendante de la langue, et a été testée sur des langues typologiquement distantes : deux langues régionales de France, l'alsacien et le picard, et une autre langue peu dotée, le malgache, une langue austronésienne.

2 État de l'art

Depuis quelques années, les architectures Bi-LSTM ont montré leur potentiel sur des tâches d'étiquetage de séquences, et en particulier d'étiquetage en parties du discours (Horsmann & Zesch, 2017). Cependant, ce type d'approche nécessite une très grande quantité de données, qui est loin des quantités disponibles dans le cas de langues peu dotées (15M de tokens pour entraîner les plongements lexicaux avec fastText et entre 50 000 et 2M de tokens annotés pour entraîner l'étiqueteur bi-LSTM). Peu d'évaluations ont été faites sur des langues peu dotées.

(Fang & Cohn, 2016) rapportent des résultats mitigés sur le malgache : ils utilisent un corpus aligné pour projeter les annotations depuis l'anglais et pour modifier le réseau de neurones afin de dépasser l'état de l'art, sans quoi les performances sont mauvaises.

Les performances de ces réseaux de neurones sont fortement dépendantes de la qualité des plongements lexicaux. L'un des modèles très largement utilisé est le modèle SkipGram, et en particulier l'outil fastText (Bojanowski *et al.*, 2016). Deux caractéristiques de fastText sont intéressantes dans le cadre de l'étiquetage en parties du discours de langues peu dotées : il prend en compte des informations internes aux mots, ce qui permet de capturer des informations morphologiques, essentielles pour l'étiquetage en parties du discours ; de plus il est capable de générer des vecteurs pour les mots hors vocabulaire, ce qui est également indispensable dans le cadre de langues peu dotées.

Concernant les langues étudiées, le cas du malgache a déjà été évoqué avec les travaux de (Fang & Cohn, 2016). Le picard et l'alsacien ont fait l'objet de travaux fondés sur une transposition de mots outils (Bernhard & Ligozat, 2013; Magistry *et al.*, 2017), et, pour l'alsacien, sur l'annotation collaborative de corpus (Millour *et al.*, 2017). Nous comparerons nos résultats aux méthodes par transposition et à notre implémentation d'un étiqueteur basé sur un modèle MaxEnt semblable à celui utilisé dans (Millour *et al.*, 2017).

3 Adaptation des plongements lexicaux au cas des langues peu dotées

Les plongements lexicaux sont souvent considérés comme un outil de représentation de la similarité syntactico-sémantique, mais en réalité, étant censés capturer des similarités distributionnelles, ils peuvent être adaptés à différents niveaux d'analyse linguistique. Une telle spécialisation apparaît même nécessaire lorsque ceux-ci sont entraînés sur de petites quantités de données afin que les similarités puissent être plus facilement capturées.

3.1 Description du système

Dans cet article, nous nous appuyons sur une architecture classique de Bi-LSTM, pour laquelle nous utilisons l'implémentation YASET (Tourille *et al.*, 2017), reprenant l'architecture de (Lample *et al.*, 2016).

Nous avons fixé les paramètres de YASET en tenant compte du fait que nos corpus sont petits et donc du risque de sur-apprentissage. Nous avons également essayé d'utiliser hyperopt mais étant donné la variété des situations étudiées, il est peu probable de trouver un jeu de paramètres qui conviennent à toutes. Nous utilisons une couche cachée de taille 30, et optimisons par *adam* avec un loss ratio de $0,001$ et un *dropout* à $0,5$.

3.2 Plongements lexicaux pour l'étiquetage en parties du discours

Dans ce travail, nous proposons une définition du contexte pour les plongements lexicaux qui est fondée sur la tâche d'étiquetage visée. Nous utilisons le modèle SkipGram avec un échantillonnage de contre-exemples (*negative sampling*) popularisé par *word2vec* (Mikolov *et al.*, 2013), et étendu ensuite dans fastText (Bojanowski *et al.*, 2016) pour prendre en compte des informations internes aux mots, en représentant les mots par leurs n-grammes de caractères.

Le système fastText cherche à prédire les mots du contexte étant donné un mot cible ou un n-gramme de caractères, pour lequel on cherche à construire un vecteur. fastText est ici utilisé comme système baseline pour construire les plongements lexicaux (voir figure 1b).

Nous proposons d'étoffer ce modèle en ne nous limitant pas aux formes avoisinantes pour définir les éléments du contexte afin de construire des plongements lexicaux spécialisés pour la tâche d'analyse en parties du discours ; ces plongements seront appelés MorphoSyntactic Embeddings (MSE).

Afin de spécialiser les plongements lexicaux, nous forçons le modèle SkipGram à s'intéresser aux indices généralement suivis pour une telle analyse (y compris lorsque celle-ci est manuelle). Nous construisons ainsi un modèle qui vise à prédire ces indices à partir d'un mot cible donné.

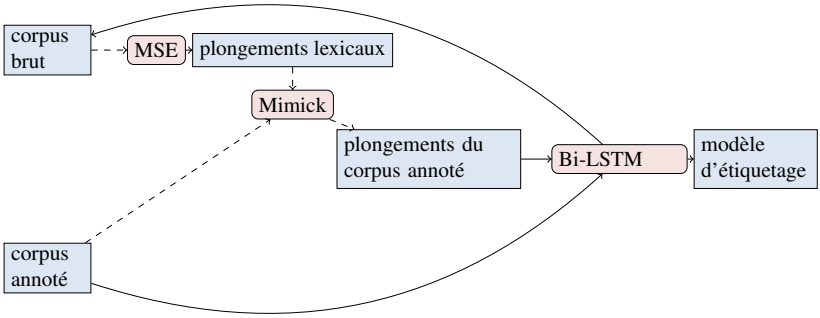
Les indices retenus sont les suivants :

— formes précédente et suivante ;
— morphèmes du mot cible ;
— morphèmes des mots précédent et suivant ;
— parties du discours des mots précédent et suivant ;
— mots grammaticaux les plus proches à droite et à gauche du mot cible.

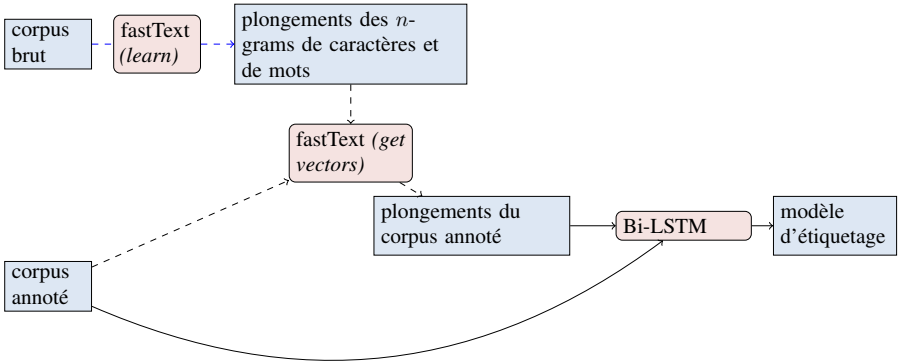
Pour obtenir les morphèmes, nous utilisons l'outil Morfessor (Virpioja *et al.*, 2013) [1]. Afin d'obtenir les informations sur les étiquettes des mots voisins et les mots grammaticaux, nous procédons de manière itérative (voir figure 1a). Une première version de notre système ignorant ces informations est utilisée pour annoter le corpus brut, et cette information est prise en compte lors d'une seconde exécution.

Un exemple d'indices considérés pour une occurrence de *Làcha* dans notre corpus alsacien est donné à la figure 2 : la première ligne correspond au mot courant avec son contexte, la seconde ligne est

1. Cet outil segmente la forme et produit donc un sous-ensemble des n-grammes qu'elle contient. En nous limitant à ce sous-ensemble, nous pensons éviter une partie du bruit qui nuit au modèle de fastText, particulièrement dans le cas de nos petits volumes de données. De plus, à la différence de fastText nous prenons en compte la morphologie des mots voisins.



(a) MSE



(b) fastText

FIGURE 1: Architectures des systèmes : avec plongements MSE et avec fastTest (les pointillés représentent des apprentissages non supervisés)

Èbbis	genschtigeres	às	Làcha	gìtt	’s	uf	dr	gànza	Walt	nìt
PRON	ADJ	SCONJ	?	VERB	PRON	ADP	DET	ADJ	NOUN	PART
ebb-is	gen-scht-iger-es	às	Làch-a	gìt-t	’s	uf	fr	gànz-a	Walt	nìt
quelque chose	plus abordable	que	rire	exister	ça	sur	le	entier	monde	pas

indices pour ‘Làcha’			
Context type	value	Context type	value
morpheme	Làch-	next-morph	gìt-
morpheme	-a	next-morph	-t
prev-form	às	next-form	gìtt
prev-tag	SCONJ	next-tag	VERB
prev-funct-word	às	next-funct-word	’s

FIGURE 2: Extraction des indices morphosyntaxiques

la séquence de parties du discours qui ont été attribuées lors de la première annotation, la troisième ligne correspond aux morphèmes de chaque mot, déterminés par *Morfessor*, et la quatrième ligne est la traduction française. À partir de ces informations, les indices extraits pour cette occurrence du mot *Làcha* sont indiqués dans le tableau : par exemple, les morphemes *Làch-* et *-a* du mot cible, la forme précédente *às* etc.

3.3 Gestion des mots inconnus

L’une des difficultés dans l’utilisation de ces modèles est la prise en compte des mots peu fréquents, voire absents du corpus d’apprentissage. Cette difficulté est d’autant plus importante lorsque les corpus disponibles sont de faible taille : lorsque les mots hors vocabulaire représentent environ la moitié des mots (ce qui est le cas dans les corpus étudiés), il n’est pas possible de leur attribuer la même représentation à tous.

Dans cet article, nous nous comparons à fastText comme méthode de base. Celui-ci crée des vecteurs pour les mots inconnus en additionnant les vecteurs qu’il a construit pour des sous-mots (n-grammes de caractères). Notre système ne construit pas de représentation pour les n-grammes, mais uniquement pour les mots cibles observés dans le corpus de données brutes. Pour générer des vecteurs pour les mots inconnus, nous utilisons le système proposé par (Pinter *et al.*, 2017), Mimick, qui entraîne un Bi-LSTM sur les caractères pour prédire des vecteurs en fonction de la graphie des mots. Mimick est entraîné sur les vecteurs déjà existants. L’un des paramètres essentiels dans cette configuration est le nombre minimal d’occurrences à partir duquel un token sera pris en compte pour le calcul des plongements lexicaux : en effet, les fréquences étant globalement assez faibles dans nos corpus, fixer un seuil trop haut éliminerait trop de mots (qui seraient cependant gérés par Mimick, mais Mimick a besoin de suffisamment de vecteurs également) ; en revanche, fixer un seuil trop bas conduit à des plongements de mauvaise qualité.

4 Validation sur le français

Nous avons tout d’abord mené des tests sur le français car il existe de nombreuses ressources pour le français, ce qui permet de contrôler la quantité de données utilisée à la fois pour créer les plongements lexicaux et pour l’entraînement de l’étiqueteur.

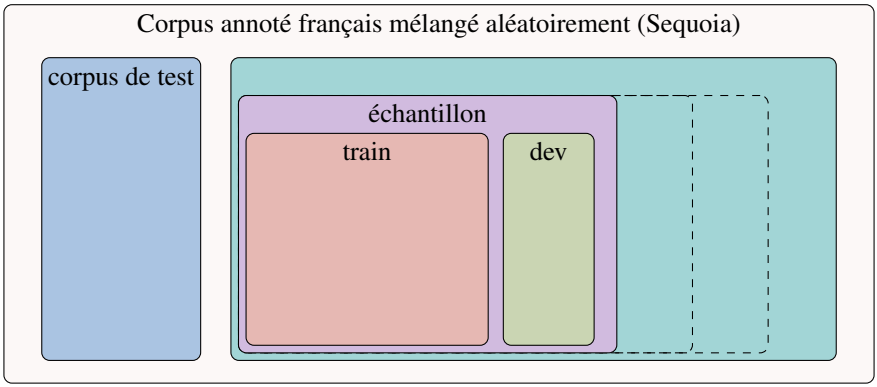


FIGURE 3: Échantillonnage des données annotées. Les tailles des corpus annotés varient entre 500 et 50, 000 tokens. Le corpus de test a été mis de côté pour servir de jeu d'évaluation pour tous les tests.

Pour ces expériences, nous avons utilisé comme données brutes un corpus de la Wikipédia française au format texte [2] et comme données annotées le corpus Sequoia (Candito & Seddah, 2012), qui a été converti vers le jeu d'étiquettes Universal POS tags.

Dans un premier temps, nous avons étudié l'influence de la quantité de données utilisée sur la qualité de l'étiquetage. Pour cela, nous avons tout d'abord mis de côté 20% du corpus Sequoia, que nous utilisons comme test pour toutes les expériences. Pour les corpus d'entraînement, nous avons mélangé les phrases et nous vérifions que les plus gros corpus incluent toujours les plus petits afin de limiter les effets de genre ou de thème (voir figure 3).

Nous avons utilisé des tailles de corpus brut allant de 200 000 tokens à 20M pour les plongements lexicaux, et de 500 à 50 000 tokens pour les données annotées. Nous évaluons notre système sur ces différentes tailles de corpus, et comparons notre méthode de construction de plongements à fastText. Les résultats sont donnés en figure 4.

Ces résultats montrent que les performances sont très bonnes lorsque les quantités de données sont importantes, à la fois en terme de corpus brut et de corpus annoté. Néanmoins, dans le cas, artificiel pour le français, de faibles volumes de données (en bas à gauche des matrices), notre système obtient de meilleurs résultats que fastText.

4.1 Plongements lexicaux

L'analyse des plongements lexicaux pour 500 000 tokens permet d'explorer le comportement des plongements crées par fastText et MSE. La figure 5 montre ces plongements après réduction de dimensionnalité à 2 avec l'outil T-SNE. Chaque point représente une forme du corpus et les couleurs représentent la partie du discours la plus souvent attribuée à cette forme. Dans la partie droite de la figure, qui correspond à nos plongements lexicaux, les parties du discours sont relativement séparées les unes des autres, contrairement à ce qui est observé avec les plongements lexicaux de fastText. C'est une explication possible de la différence de score de YASET lorsqu'il est utilisé avec l'une ou l'autre des représentations.

2. Mis à disposition par `https://ufal.mff.cuni.cz/w2c`

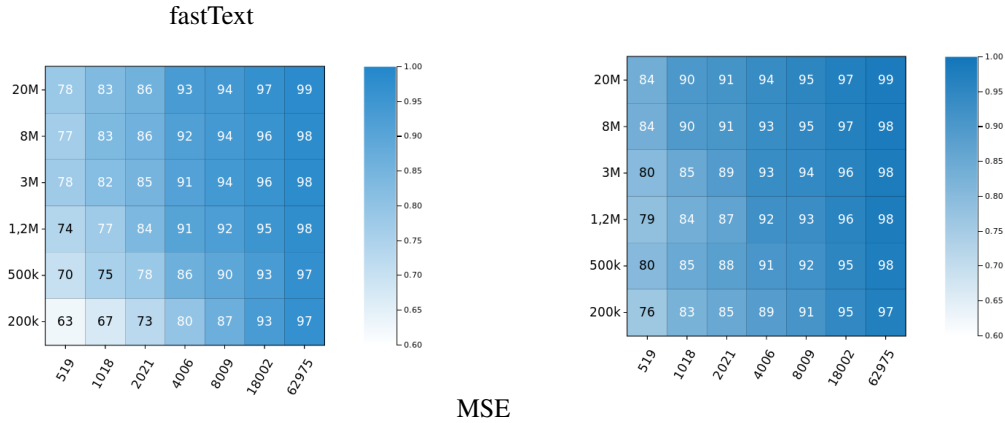


FIGURE 4: Scores d'étiquetage en parties du discours obtenus sur le corpus Sequoia en utilisant fastText ou nos embeddings (MSE), en faisant varier la taille du corpus annoté (abscisse) et du corpus brut (ordonnée)

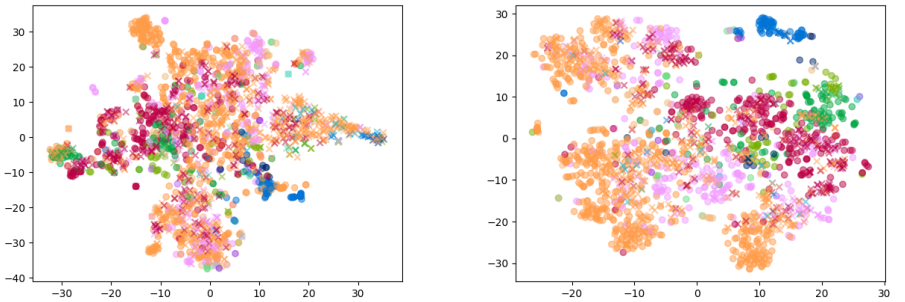


FIGURE 5: Plongements lexicaux obtenus avec fastText et avec MSE. Chaque point représente un mot, et chaque couleur une partie du discours

5 Expériences sur des langues peu dotées

Notre objectif étant de créer une méthode d'étiquetage en parties du discours pour des langues peu dotées, nous avons testé notre système sur trois langues peu dotées : deux langues régionales de France, le picard et l'alsacien, et le malgache, qui nous permet de comparer nos travaux à un état de l'art récent.

	données		exactitude			
langue	brutes	annotées	MSE	fastText	transposition	MaxEnt
alsacien	200 000	12 600	**0,91**	0,86	0,78	0,85
picard	1,9M	9 640	**0,89**	0,82	0,71	0,78
malgache	2M	4 230	**0,91**	0,84	n/a	0,86
malgache (Fang & Cohn 2016)			0,87			

TABLE 1: Scores d'étiquetage en parties du discours

5.1 Langues étudiées

Le malgache est une langue de la famille des langues austronésiennes. C'est une langue officielle à Madagascar où elle est parlée par plus de 20M de personnes. Il s'agit d'une langue de type VOS, morphologiquement riche, de type agglutinative.

L'alsacien est une langue parlée dans le nord est de la France, principalement en Alsace. Elle est composée de plusieurs dialectes, qui pour la plupart sont issus des langues alémaniques et franciques. Si l'alsacien présente de nombreuses variantes inter- et intra-dialectales, les principales caractéristiques morphologiques se retrouvent dans toutes ces variantes. Pour l'essentiel, on peut retenir que les verbes reçoivent des morphèmes de temps, mode et nombre et les substantifs des morphèmes de nombre, cas et genre.

Le picard est une langue d'oïl, qui appartient à la famille des langues romane. La zone géographique du picard couvre la région des Hauts-de-France et la province de Hainaut en Belgique. Bien que proche du français, le picard présente des particularités. Ainsi, l'ordre des mots peut être différent. Par exemple, *il o foait keud assé* se traduit en *il fait assez chaud* où l'on peut constater que une inversion de l'ordre entre l'adverbe (assé/assez) et l'adjectif (keud/chaud). De plus le picard présente de nombreuses contractions de prépositions et déterminants (par exemple *d'ches, su, al*) et des néologismes construits par composition.

5.2 Corpus

Les corpus bruts utilisés ont des tailles de l'ordre de 2 millions de tokens pour le malgache et le picard, et de 200 000 tokens pour l'alsacien (voir tableau 1). Le corpus malgache est composé d'articles du site *Global Voice* [3], également utilisés par (Fang & Cohn, 2016). Pour le picard, la base textuelle Picartext constitue la source des données brutes, et pour l'alsacien, ce sont les articles de la Wikipédia alémanique annotés comme étant en alsacien [4].

En ce qui concerne les données annotées les corpus du projet RESTAURE ont été utilisés [5]. Le corpus annoté pour l'alsacien contient environ 12 600 tokens, pour le picard environ 9 700 tokens, tandis que celui pour le malgache est à environ 4 200. Ces situations correspondent environ au milieu des matrices de la figure 4. Cependant, les deux langues régionales de France étudiées sont moins normalisées que le français, donc on peut s'attendre à de moins bons scores avec une méthode standard.

3. https://www.cs.cmu.edu/~ark/global-voices/
4. La Wikipédia alémanique contient des articles dans plusieurs dialectes correspondant à l'aire linguistique alémanique
5. https ://zenodo.org/communities/restaure

Tous les corpus annotés utilisent les Universal POS tags [6].

Pour le malgache, nous avons gardé la division train/test de (Fang & Cohn, 2016). Pour les deux langues régionales de France, les expériences ont été menées en validation croisée en 5 tirages après mélange aléatoire des phrases du corpus.

5.3 Résultats

Les résultats obtenus avec les différents systèmes sont présentés dans le tableau 1.

Les scores obtenus avec notre système avec les plongements MSE sont systématiquement supérieurs à ceux obtenus en utilisant fastText : 0,91 contre 0,86 en alsacien ; 0,89 contre 0,82 en picard ; et 0,91 contre 0,84 en malgache, ce qui nous situe bien au-dessus de l'état de l'art de (Fang & Cohn, 2016), sans utiliser de données bilingues.

Nous avons également indiqué les scores obtenus par une méthode de transposition proche de celle de (Bernhard & Ligozat, 2013), qui se fonde sur des étiqueteurs de langues proches : modèle allemand du Stanford POS Tagger (Toutanova *et al.*, 2003) pour l'alsacien, modèle MaxEnt français entraîné sur Sequoia pour le picard ; le malgache ne possède pas à notre connaissance de langue proche bien dotée. Cette méthode de transposition obtient des résultats bien inférieurs, bien qu'au-dessus de ceux de fastText en alsacien.

Enfin, la dernière colonne donne les scores d'un système très proche de MElt (Denis & Sagot, 2009), utilisé par (Millour *et al.*, 2017), et dont les performances sont également significativement en-dessous de notre système.

6 Conclusion

Dans cet article, nous avons présenté une nouvelle méthode d'étiquetage en parties du discours pour les langues peu dotées, fondée sur une adaptation des plongements lexicaux. Les résultats obtenus dépassent l'état de l'art pour chaque des langues étudiées. La prise en compte des mots hors vocabulaire pourrait cependant être encore améliorée, notamment en prenant en compte leur contexte pour générer leur vecteur.

Remerciements

Ces travaux ont bénéficié du soutien de l'ANR (projet RESTAURE - référence ANR-14-CE24-0003).

6. http://universaldependencies.org

Références

BERNHARD D. & LIGOZAT A.-L. (2013). Hassle-free POS-Tagging for the Alsatian Dialects. In *Special volume on 'Non-Standard Data Sources in Corpus Based-Research'*.

BOJANOWSKI P., GRAVE E., JOULIN A. & MIKOLOV T. (2016). Enriching Word Vectors with Subword Information. *arXiv preprint ArXiv :1607.04606*.

CANDITO M. & SEDDAH D. (2012). Le corpus sequoia : annotation syntaxique et exploitation pour l'adaptation d'analyseur par pont lexical (the sequoia corpus : Syntactic annotation and use for a parser lexical domain adaptation method) [in french]. In *Proceedings of the Joint Conference JEP-TALN-RECITAL 2012, volume 2 : TALN*, p. 321–334 : ATALA/AFCP.

DENIS P. & SAGOT B. (2009). Coupling an annotated corpus and a morphosyntactic lexicon for state-of-the-art POS tagging with less human effort. In *Proceedings of the 23rd Pacific Asia Conference on Language, Information and Computation, Volume 1*, volume 1.

FANG M. & COHN T. (2016). Learning when to trust distant supervision : An application to low-resource POS tagging using cross-lingual projection. In *Proceedings of the 20th SIGNLL Conference on Computational Natural Language Learning, CoNLL 2016, Berlin, Germany, August 11-12, 201 6*, p. 178–186.

HORSMANN T. & ZESCH T. (2017). Do LSTMs really work so well for PoS tagging ?–A replication study. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, p. 727–736.

LAMPLE G., BALLESTEROS M., SUBRAMANIAN S., KAWAKAMI K. & DYER C. (2016). Neural architectures for named entity recognition. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics : Human Language Technologies*.

MAGISTRY P., LIGOZAT A.-L. & ROSSET S. (2017). Expériences d'étiquetage morphosyntaxique dans le cadre du projet RESTAURE. In *Atelier Diversité linguistique et TAL (DiLiTAL 2017) associé à la conférence TALN*.

MIKOLOV T., SUTSKEVER I., CHEN K., CORRADO G. S. & DEAN J. (2013). Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, p. 3111–3119.

MILLOUR A., FORT K., BERNHARD D. & STEIBLE L. (2017). Toward a lightweight solution to the language resources bottleneck issue : creating a POS tagger for Alsatian using voluntary crowdsourcing . In *Traitement Automatique des Langues Naturelles (TALN)*, Orléans, France.

PINTER Y., GUTHRIE R. & EISENSTEIN J. (2017). Mimicking Word Embeddings using Subword RNNs. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, p. 102–112.

TOURILLE J., FERRET O., NÉVÉOL A. & TANNIER X. (2017). Neural Architecture for Temporal Relation Extraction : A Bi-LSTM Approach for Detecting Narrative Containers. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2 : Short Papers)*, p. 224–230, Vancouver, Canada : Association for Computational Linguistics.

TOUTANOVA K., KLEIN D., MANNING C. D. & SINGER Y. (2003). Feature-rich part-of-speech tagging with a cyclic dependency network. In *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology-Volume 1*, p. 173–180 : Association for Computational Linguistics.

VIRPIOJA S., SMIT P., GRÖNROOS S.-A. & KURIMO M. (2013). *Morfessor 2.0 : Python Implementation and Extensions for Morfessor Baseline.* Rapport interne, Helsinki.

Modélisation des processus d'acquisition syntaxique par jeux de langage entre agents artificiels

Marie Marcia[1] Isabelle Tellier[1]
(1) Lattice (UMR 8094), CNRS, ENS Paris, Université Sorbonne Nouvelle,
PSL Research University, USPC, 1 rue Maurice Arnoux, 92120 Montrouge, France
marie.marcia@univ-paris3.fr, isabelle.tellier@univ-paris3.fr

RÉSUMÉ

Dans cet article, nous présentons une modélisation de la situation d'acquisition de la syntaxe de sa langue maternelle par un enfant inspirée des "jeux de langages" de Luc Steels. Le modèle suppose que l'enfant a accès à une représentation sémantique des énoncés qui lui sont adressés, et qu'il doit réagir en désignant la tête syntaxique de ces énoncés. Nous décrivons des expériences exploitant des données du corpus CHILDES et mettant en jeu un processus d'acquisition simple mais efficace.

ABSTRACT

Modeling Syntactic Acquisition by Language Games between Artificial Agents

In this paper, we present a model of the way a child acquires her mother tongue syntax, inspired by Luc Steels' "language games". The model assumes that the child has access to the semantics of the utterances addressed to her, and that she must react by designing the syntactic head of these utterances. We describe experiments made with data from the CHILDES database, using a simple but effective acquisition process.

MOTS-CLÉS : analyse syntaxique, sémantique, modélisation de l'acquisition, jeux de langage.

KEYWORDS: parsing, semantics, model of language acquisition, language games.

1 Introduction

Nous proposons dans cet article un modèle d'apprentissage automatique de la syntaxe d'une langue naturelle qui s'inspire de l'acquisition humaine. Ce modèle s'appuie sur le paradigme des jeux de langage, qui visent à modéliser l'émergence de connaissances linguistiques dans une population de locuteurs artificiels. Le cadre proposé ici fait intervenir non pas une population mais seulement deux locuteurs artificiels : un expert (ou adulte) et un apprenant (ou enfant) qui doit acquérir, au fil d'interactions avec l'expert, une compétence syntaxique. Les composantes phonétique et lexicale de l'apprentissage de la langue sont considérées comme déjà acquises [1] et n'entreront pas en ligne de compte. La sémantique joue en revanche un rôle fondamental. Notre modèle s'appuie sur des capacités sémantiques conçues comme un prérequis nécessaire à l'acquisition syntaxique.

La section suivante définit le paradigme des jeux de langage dans lequel se place notre modèle. Nous

1. Il ne s'agit pas de suggérer qu'à l'âge où l'enfant commence à acquérir des compétences syntaxiques, les compétences phonétique et lexicale sont complètement acquises. Cependant, la tâche proposée portera uniquement sur l'acquisition de la syntaxe, où la connaissance lexicale notamment est un prérequis.

détaillerons ensuite le protocole expérimental mis en place : le corpus utilisé, le déroulement du jeu de langage défini, les méthodes de parsing et d'apprentissage automatique mis en œuvre. Enfin, les résultats des premières expériences menées seront analysés et évalués.

2 Jeux de langage

Une approche originale de l'acquisition de connaissances linguistiques a été proposée par le paradigme des jeux de langage (Steels, 1995; Kaplan, 2001). Ce paradigme, au sein de la linguistique dite évolutionnaire (Kirby, 2002; Dessalles, 2000), tente de rendre compte des processus pouvant mener à l'émergence et à l'évolution des langues naturelles, en proposant des expérimentations logicielles et robotiques. Puisqu'il est difficile d'observer empiriquement l'émergence d'une nouvelle langue, les chercheurs utilisant les jeux de langage suggèrent de recréer artificiellement les conditions d'émergence d'un système linguistique : une communauté de locuteurs potentiels, un objectif communicatif et des sens à exprimer. À partir de ces éléments, ils proposent des mécanismes pouvant mener à l'émergence d'un lexique commun.

L'intérêt majeur du paradigme des jeux de langage est de trouver un équilibre entre la simplicité de la modélisation et la complexité de l'interaction langagière située (Wellens, 2012). Ce paradigme repose sur la simulation d'interactions langagières simplifiées entre binômes de locuteurs artificiels (robotiques ou logiciels), qui tentent de s'exprimer (et de se comprendre) à propos d'un contexte d'objets (réels ou également artificiels). On peut comparer une simulation de ce type à l'émergence d'un protolangage (Bickerton, 1990) dans une population humaine, qui correspond à un stade pré-syntaxique de la langue où s'établit une correspondance entre des unités lexicales et des unités sémantiques. Ce type d'émergence et d'évolution au sein d'un groupe est à mettre en parallèle avec l'acquisition de la langue par l'enfant, qui connaît également ce stade pré-syntaxique.

Le paradigme des jeux de langage s'est surtout focalisé jusqu'à présent sur l'acquisition de connaissances lexicales. Plus récemment (Steels & Garcia-Casademont, 2015), il a été étendu à des jeux de langage syntaxiques. Il s'agit alors, à partir d'une simulation, de faire émerger une grammaire commune chez une population d'agents disposant déjà de capacités lexicales et d'une représentation sémantique du monde. Cette grammaire émerge, là encore, grâce à une succession d'interactions des locuteurs s'exprimant à propos de leur monde. La grammaire se construit peu à peu par la réalisation de jeux de langage consistant en la production et l'interprétation d'énoncés référant à des objets ou des événements du monde simulé.

On trouve dans ce paradigme l'essentiel des paramètres que nous voulons réaliser dans notre propre modélisation : un modèle évolutif basé sur l'interaction, l'utilisation d'exemples positifs de la langue à acquérir et la composante sémantique nécessaire à l'acquisition d'une grammaire. Mais notre modèle se distingue de celui de Steels par le formalisme syntaxico-sémantique utilisé et le nombre d'agents intervenant dans l'interaction : une population chez Steels, une simple paire "expert/apprenant" pour nous. Notre modélisation cherche en effet à simuler l'acquisition individuelle de la langue par un enfant et non l'émergence d'une langue nouvelle dans un groupe.

3 Protocole expérimental

3.1 Corpus

Le corpus que nous avons constitué pour l'input fourni par l'expert à l'apprenant est extrait d'une collection des corpus français de la base de données de la plate-forme CHILDES (MacWhinney, 2000). Ces textes sont la transcription de discussions entre des enfants (cibles d'études parfois longitudinales) et d'autres interlocuteurs (les parents, les linguistes ou des pairs). Ces données sont particulièrement adaptées à notre tâche puisqu'elles constituent un corpus de Français oral [2] qui comporte des milliers de phrases adressées (pour la majorité) par des adultes à des enfants.

Nous avons extrait du corpus tous les énoncés produits par les participants autres que l'enfant cible, puis filtré ces énoncés pour qu'il ne reste plus que ceux syntaxiquement analysables. Le parser utilisé pour cela est Grew (Guillaume & Perrier, 2012), qui présente l'intérêt de produire aussi une représentation sémantique des énoncés (cf. section 3.2). Le critère minimaliste utilisé est que le parser renvoie un seul graphe de dépendances, donc que la phrase ne comporte qu'une seule racine. Les autres phrases, non analysables par le parser, sont éliminées. Avec ce tri, environ 44% des phrases sont éliminées. Le parser Grew est en effet conçu pour traiter du Français écrit et non oral. La proportion de phrases analysables est cependant amplement suffisante pour notre corpus.

La Table 1 donne le nombre de phrases prononcées par des adultes extraites du corpus en fonction de l'âge de l'enfant destinataire. Les proportions étant très disparates, le corpus a été échantillonné pour obtenir une égale proportion d'inputs présentés à des enfants de 0 à 1 an (11 mois et 17 jours pour le plus jeune), de 1 à 2 ans, etc, jusqu'à 6-7 ans (6 ans 11 mois et 26 jours pour le plus âgé) : 400 énoncés par tranche d'âge.

Âge de l'enfant cible	0	1	2	3	4	5	6
Nombre de phrases	745	55363	102688	44244	15062	6283	2621
Longueur moyenne des phrases	4.57	4.88	5.61	5.68	6.04	5.93	5.7

TABLE 1 – Nombre de phrases et longueur moyenne des phrases en fonction de l'âge de l'enfant destinataire dans le corpus non échantillonné

Le tableau donne aussi la longueur moyenne des phrases adressées aux enfants, qui augmente en fonction de leur tranche d'âge [3]. Les 2800 énoncés constituant le corpus échantillonné ont donc été triés pour apparaître, pendant nos expériences, par ordre croissant d'âge de l'enfant destinataire.

Nous donnons pour information la représentation des différentes catégories morphosyntaxiques (selon le jeu d'étiquettes morphosyntaxiques de MElt (Denis & Sagot, 2009) utilisé ici) parmi les racines syntaxiques de la totalité des énoncés de notre corpus. Il s'agit pour la majorité de racines verbales, mais on trouve également une proportion considérable de noms, prépositions et conjonctions (de coordination ou de subordination) :

— Verbes : 64.11%

2. Voici quelques exemples de phrases que l'on pourra trouver dans notre corpus : "encore une banane", "ah bah le hérisson", "ah hop", "tu me disais", "non non non", "pas prendre le cube d'Ulysse", "ohlàlà Julie".

3. La longueur des énoncés est certes à mettre en lien avec l'âge de l'enfant destinataire, mais la façon de parler des individus est évidemment variable et influe fortement sur cette longueur moyenne. Le type de dialogue peut également avoir une influence : le discours des parents à leur enfant est plutôt spontané, tandis que certaines questions posées à l'enfant par le chercheur peuvent être préparées, et éventuellement plus longues.

— Noms : 12.11%
— Prépositions : 9.82%
— Conjonctions : 8.82%
— Mots étrangers [4] : 2.07%
— Adverbes : 1.07%
— Pronoms : 0.93%
— Adjectifs : 0.57%
— Prépositions + déterminants : 0.50%

3.2 Déroulement du jeu de langage

Dans notre modèle interviennent seulement deux interlocuteurs artificiels, un expert et un apprenant (ou bien un adulte et un enfant). L'apprenant n'a initialement aucune connaissance syntaxique, mais il a des connaissances lexicales. Il est donc capable de comprendre le sens de mots isolés. Considérer la connaissance lexicale comme acquise est un parti pris qui se base premièrement sur le fait que, chronologiquement, le processus d'acquisition d'une compétence lexicale chez l'enfant intervient avant le début du processus d'acquisition syntaxique. Il existe un chevauchement dans l'acquisition de ces deux compétences, mais la compétence syntaxique permettant de comprendre et de former des "phrases" de plus d'un mot apparaît bel et bien après l'étape de l'acquisition lexicale qui permet à l'enfant d'associer un sens à un nombre de mots encore limité (c'est-à-dire vers l'âge de 2 ans). On donne dans le tableau suivant, tiré de (Tellier, 2005) la chronologie de l'acquisition des différentes compétences linguistiques. D'autre part, notre travail se focalise sur l'acquisition de la syntaxe : l'apprentissage du lien entre un mot et son sens est une tâche distincte que nous préférons écarter en le considérant comme acquis.

Au cours d'un jeu, donc d'une interaction, l'expert "prononce" un énoncé (provenant du corpus échantillonné). L'apprenant a accès à cet énoncé et à sa représentation sémantique globale (sous forme de graphe [5]). On considère en effet qu'un énoncé est toujours prononcé dans un contexte dans lequel il peut être compris. Le graphe sémantique de l'énoncé simule ce contexte virtuel, réduit au minimum puisque limité à cet énoncé. L'apprenant est donc exposé à un énoncé au sein duquel il est capable de faire le lien entre un mot et son sens lexical, ainsi qu'au sens global de cet énoncé. Mais il n'a pas accès à sa structure syntaxique.

Le but du jeu est, pour l'apprenant, de désigner, dans son environnement (le contexte, représenté par le graphe sémantique), le sens (donc le nœud sémantique) exprimé par la racine syntaxique de l'énoncé. Ce principe s'inspire d'environnements simulés (par exemple SHRDLU (Winograd, 1971)) où les objets sont désignés par un terme qui est la racine d'une phrase nominale ("le triangle bleu sur le cube rouge"). Nous étendons ce principe à des énoncés oraux où la racine peut désigner autre chose qu'un objet. La notion de désignation devient alors plus conceptuelle. Dans une phrase comme "Elle est là la maison du singe", l'apprenant doit "désigner" le sens correspondant au verbe.

Pour réussir le jeu, l'apprenant doit acquérir une compétence syntaxique. Une fois que l'enfant a donné sa réponse, l'expert confirme ou infirme mais ne donne pas la solution. Le but n'est pas en effet que l'enfant apprenne à associer une bonne réponse à une phrase donnée, mais qu'il apprenne à

4. Cette catégorie recouvre en fait à la fois les mots étrangers, mais aussi les onomatopées et marqueurs de l'oral comme "tchou", "bah", "ben", "okay", "hou", etc.

5. Le graphe sémantique est un graphe orienté, connecté, acyclique, et n'a pas de racine. On peut voir en Figure 3 qu'un nœud sémantique peut aussi bien n'avoir aucun gouverneur qu'en avoir un ou plusieurs.

âge	capacités phonétiques	capacités lexicales	capacités syntaxiques
6-15 sem.	début du *babil*		
3-8 mois	*babil* riche		
1 an	le *babil* s'estompe ; qq. exclamations	4-5 *fonctions* pour les exclamations	
1 an 1/2	pauvreté (contrastant avec le *babil*)	30 à 50 mots : noms, adjectifs, verbes d'action	*holophrases* (phrases à un mot)
2 ans	lente amélioration : état provisoire	50 à quelques centaines de mots	*stlyle télégraphique* (phrases à 2 mots)
2 ans 1/2	idem	700 à 800 mots (proportion de noms 4 fois supérieure à celle de l'adulte)	phrases à 3 mots et plus ; nombreuses fautes
3 ans	presque adulte	un millier et plus	phrases bien formées
4 ans	quasi adulte	proche de l'adulte : env. 3000 mots (adulte : 10000 mots)	proche de l'adulte

FIGURE 1 – Chronologie de l'acquisition des compétences linguistiques

induire une structure syntaxique à partir de n'importe quelle phrase. L'expérience consiste en une série d'interactions de ce type, qui testent et simulent uniquement la capacité de compréhension.

La réussite d'un jeu de désignation en compréhension est soumise à la nécessité de tenter une analyse de la structure syntaxique de l'énoncé. On pourrait créer un jeu de désignation en production, où les rôles seraient inversés : l'apprenant devrait produire des énoncés et l'enseignant en désigner la racine syntaxique. Si l'enseignant parvient à trouver la bonne racine, c'est que l'énoncé est suffisamment grammatical. L'intérêt en termes d'apprentissage serait moindre puisqu'il s'agirait pour l'apprenant de produire une séquence de mots, et non une structure syntaxique. Le feedback de l'enseignant apporterait également peu à l'apprenant : soit l'enseignant confirme ou infirme la grammaticalité de la séquence (ce qui n'apporte pas de connaissance syntaxique à l'apprenant), soit il fournit une correction (or ce n'est pas l'objectif de notre modélisation de donner une correction ou "feedback négatif"). De manière générale, notre dispositif se prête mal à des jeux de production par l'apprenant : si le jeu réussit, il n'apprenant rien, s'il échoue, il obtient une éventuelle correction.

D'autres types de jeux de compréhension sont cependant envisagés pour la suite de ce travail : le premier est un jeu, non plus de désignation, mais dans lequel l'apprenant doit déterminer la valeur de vérité d'un énoncé produit par l'adulte ; le second introduit des phrases interrogatives produites par l'enseignant et auxquelles l'apprenant doit répondre par oui ou par non en fonction du contexte.

La connaissance de l'expert (l'analyse syntaxique juste) et le graphe sémantique d'une phrase sont obtenus grâce à un outil de parsing par réécritures de graphes, Grew. Celui-ci permet, à partir d'une phrase étiquetée en POS (ici par MElt (Denis & Sagot, 2009)), d'obtenir un graphe de dépendances de cette phrase, ainsi que sa représentation sémantique en DMRS (Dependency Minimal Recursion Semantics) (Copestake, 2009), représentation dont l'exigence est "d'avoir une annotation lisible et

minimale" (Guillaume & Perrier, 2012).

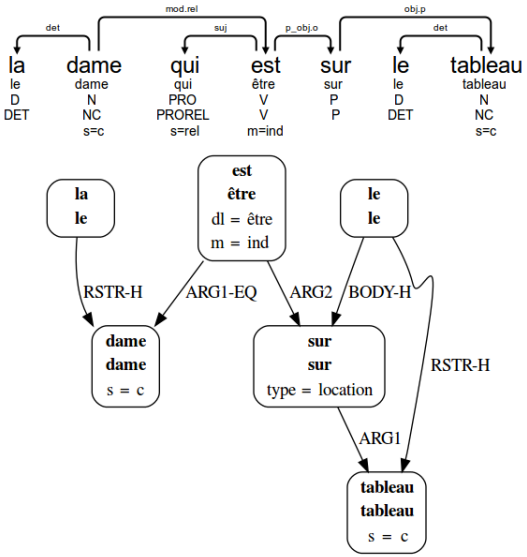


FIGURE 2 – Analyse en dépendances (en haut) et représentation sémantique (en bas) d'un énoncé du corpus réalisées grâce au parser Grew

Les Figures 2 et 3 donnent des exemples d'énoncés du corpus avec leur analyse en dépendances et leur graphe sémantique associés. Dans le cas de la Figure 2, l'expert prononce l'énoncé "la dame qui est sur le tableau". L'apprenant, après analyse de l'énoncé, doit désigner le sens correspondant à la racine syntaxique de l'énoncé. La racine étant le token "dame", l'apprenant doit désigner le noeud sémantique correspondant à ce token. On simule par ce procédé un jeu de désignation d'objet où l'enfant doit comprendre de quel objet on parle, et le désigner. Dans une expérience comme SHRDLU, l'apprenant aurait ici la possibilité de désigner la dame ou le tableau. Dans notre expérience, il a la possibilité de désigner n'importe lequel des sens exprimés dans le graphe sémantique ("la", "est", "le", "dame", "sur" ou "tableau"). Dans la Figure 3, on applique le même principe, qui devient alors plus conceptuel, puisque l'apprenant doit à nouveau désigner le sens correspondant à la racine syntaxique de l'énoncé, le token "est", or ce token ne désigne plus un objet. Si la tâche de désignation devient plus conceptuelle, elle permet en revanche de ne pas avoir à trier le corpus pour n'en conserver que des phrases nominales (qui correspondent mieux à une tâche de désignation classique). Ainsi, l'apprenant est exposé à un input plus varié et plus réaliste dans une tâche d'apprentissage de la syntaxe. Des énoncés comme "ohlàlà Julie" ou "encore une banane" (présents dans notre corpus) se prêtent moins à une tâche de désignation, mais sont bien des énoncés auxquels des enfants ont été exposés, et qui ont participé à leur acquisition de la syntaxe du Français. Le biais du jeu de désignation est un moyen de donner un objectif à l'interaction, objectif qui ne peut être atteint que par l'intermédiaire de l'apprentissage de la syntaxe.

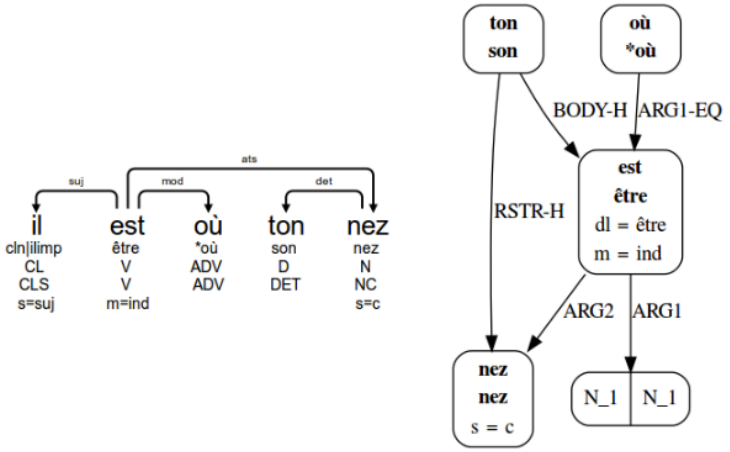


FIGURE 3 – Analyse en dépendances (à gauche) et représentation sémantique (à droite) d'un énoncé du corpus réalisées grâce au parser Grew

3.3 Parsing

Dans notre modélisation, l'analyse d'un énoncé par l'apprenant est produite par un parser en dépendances de type shift-reduce inspiré de (Yamada & Matsumoto, 2003). On modélise donc l'acquisition par l'apprentissage automatique non pas d'une grammaire (comme dans (Tellier, 2005)) mais d'un parser. Le parsing par shift-reduce revient à opérer une succession de classifications, tout en gérant une pile. C'est un modèle simple qui peut être acquis incrémentalement, et donc une hypothèse cognitivement raisonnable pour l'acquisition. Le classifieur traite des paires de tokens de la phrase et leur associe une classe. Cette classe correspond à une décision syntaxique : soit on ne met aucun lien de dépendance entre les deux tokens et on passe à la paire suivante (classe No Link), soit on met un lien de dépendance vers la gauche et on supprime le token en haut de la pile (classe Left Arc), soit on crée un lien vers la droite et on passe à la paire suivante (classe Right Arc). Le type de shift-reduce choisi ici permet d'établir des liens de dépendance non étiquetés, qui ne présupposent donc aucune connaissance syntaxique de la part de l'apprenant.

Quand, au cours du shift-reduce, toutes les paires disponibles dans la pile ont déjà été analysées, on stoppe le shift-reduce, et on analyse à la suite toutes les autres paires de tokens possibles de la phrase qui n'ont pas encore été analysées, et dont les deux éléments sont encore présents dans la pile (dans le shift-reduce, si un élément a été "réduit", donc supprimé de la pile, c'est qu'il a déjà un gouverneur et qu'il ne peut pas avoir de dépendants).

Pour classer les paires de tokens (n1, n2), l'apprenant analyse la séquence de l'énoncé et sa représentation sémantique, et construit un vecteur. Les traits choisis pour la représentation vectorielle des paires de tokens sont : l'écart entre les positions des deux tokens dans la phrase, leurs lemmes, ceux des contextes gauche et droit de n1 et n2 ; de quels arguments les nœuds sémantiques correspondant aux tokens n1 et n2 sont les prédicats, et quels sont les types de ces liens ; de quels prédicats ils sont les arguments, et le type de ces liens ; si les nœuds correspondant à n1 et n2 sont sémantiquement liés,

par un lien vers la gauche ou vers la droite. L'utilisation de l'étiquetage en POS comme trait de la représentation vectorielle, prépondérante dans les tâches d'apprentissage d'un analyseur syntaxique, est exclue ici puisque l'apprenant ne dispose d'aucune information syntaxique a priori.

Lorsqu'à un token ne correspond pas de nœud sémantique (ce qui concerne certains mots grammaticaux), les traits du vecteur correspondant aux informations sémantiques liées à ce token ont des valeurs nulles. L'absence d'information sémantique dans la représentation vectorielle de ce type de token étant commune aux mots grammaticaux, on obtient donc une classe de vecteurs particuliers (représentant une paire de tokens dont l'un ou les deux sont des mots grammaticaux) qui auront entre eux une distance vectorielle réduite. En effet, l'apprentissage des mots grammaticaux se distingue de celui des mots lexicaux : en l'absence de sens associé aux mots grammaticaux, c'est plutôt leur contexte d'apparition (dans la séquence de l'énoncé) qui va servir à leur intégration dans l'acquisition des structures syntaxiques.

Une fois la paire analysée et représentée sous la forme d'un vecteur, il faut ensuite choisir sa classe. Cette décision est prise par classification automatique avec l'algorithme des k plus proches voisins (avec une valeur de k à optimiser). Cette technique présente plusieurs avantages : elle est incrémentale et ne nécessite que peu de calcul. Très simple, elle pose peu d'hypothèses sur l'acquisition. Elle permet en outre d'utiliser des données à la fois symboliques et numériques, et ne nécessite pas de connaître à l'avance le vocabulaire présent dans l'input. Une fois que la décision par knn est prise, on vérifie que le lien à créer ne va pas donner un second gouverneur à un token.

Une fois la phrase entièrement analysée par le parser courant de l'apprenant, on choisit une racine au hasard (s'il y en a plusieurs) parmi celles produites. On la compare alors avec celle de l'enseignant (analysée avec le parser de Grew) qui donne un feedback (bonne ou mauvaise réponse). Enfin, on met à jour les données d'entraînement. On traite d'abord les nouvelles données issues de l'énoncé en entrée. Une donnée est un vecteur qui contient des informations sur une paire de tokens (avec les informations sémantiques associées) : ce vecteur se voit attribuer un score de confiance pour chaque classe. Si la donnée a déjà été rencontrée, on met à jour les scores des classes par inhibition latérale (voir paragraphe suivant). Si la donnée n'a jamais été rencontrée, on initialise les scores des classes en fonction du feedback (si le feedback est positif, les décisions qui ont été prises lors du parsing ont un meilleur score que les décisions qui n'ont pas été retenues, et inversement). On met ensuite à jour les scores des vecteurs proches des entrées rencontrées, par inhibition latérale.

La notion d'inhibition latérale est issue de la neurobiologie (Hartline *et al.*, 1956) et réfère à la capacité d'un neurone actif à réduire l'activité de ses voisins (Wellens, 2012)). Elle a été introduite dans les jeux de langage par (Steels & Kaplan, 1999). Dans un jeu de langage comme le "Naming Game" (Steels, 2000), au cours duquel un groupe d'agents artificiels doit inventer des mots pour référer à des objets puis converger vers un lexique commun pour désigner ces objets au fil d'interactions par binômes d'agents, la stratégie de l'inhibition latérale est efficace pour résoudre le problème de la concurrence entre plusieurs mots pour désigner un même objet. Plus un agent entend, au cours de ses interactions successives, un mot *m* pour désigner un objet, plus le score exprimant la certitude que le mot *m* désigne bien cet objet va augmenter, tandis que le score des mots concurrents pour désigner cet objet va décroître. Ici, la mise à jour des scores par inhibition latérale consiste, dans le cas de la réussite du jeu, à augmenter le score de la classe choisie et diminuer les scores des autres classes (et inversement dans le cas d'un échec du jeu), en fonction du feedback de l'enseignant. Autrement dit, si le jeu réussit, l'apprenant renforce sa certitude quant aux décisions syntaxiques qui ont été prises au cours de l'interaction, tandis que décroît sa certitude quant aux décisions qu'il n'a pas prises (et vice-versa en cas d'échec du jeu). Les formules de mise à jour des scores provenant de (Wellens,

2012) font intervenir les paramètres de renforcement *r* et d'inhibition *i*, tous deux fixés à 0.3 dans nos expériences (mais optimisables), et le score initial *s* d'une classe pour un vecteur. On utilise cette formule pour renforcer un score : $s + r \,.\, (1 - s)$; et la suivante pour inhiber un score : $s - i \,.\, s$. Les socres sont toujours compris entre 0 et 1.

4 Résultats et analyses

4.1 Mesure de réussite communicative

La mesure de réussite communicative est propre aux jeux de langage. Elle se calcule en divisant le nombre de fois où le jeu est un succès par le nombre total de jeux effectués (on obtient donc un score compris entre 0 et 1). Cependant, la réussite communicative gage que l'apprenant a trouvé la racine de la phrase, mais pas que le graphe de dépendance qu'il a construit pour y parvenir est correct.

Pour nos expériences, chaque séquence de jeux de langage est effectuée dix fois. La valeur optimale trouvée pour k est 5. On donne en Figure 4 les courbes moyenne, maximale et minimale de l'évolution de la réussite communicative. Deux séries d'expériences sont ici présentées : dans l'une, le corpus d'entrées est trié par âge de l'enfant destinataire ; dans l'autre, il est trié par longueur des phrases. Le tri des phrases par âge de l'enfant destinataire permet une modélisation plus réaliste. Le discours des parents étant dans une certaine mesure adapté à l'âge de l'enfant destinataire, les phrases ont donc une complexité croissante, mais cette difficulté est relative plus au lexique qu'à la syntaxe. D'un point de vue formel, le lexique n'étant pas l'objet de l'apprentissage, cette évolution de la difficulté n'a donc pas d'impact sur l'apprentissage du classifieur. Trier le corpus par longueur de phrases, de manière certes artificielle du point de vue de la modélisation, permet néanmoins de donner un critère formel qui introduit une difficulté croissante dans l'apprentissage du classifieur, et améliore ses résultats.

Lorsque le corpus est trié par âge de l'enfant destinataire, la courbe de réussite croît jusqu'à se stabiliser au-dessus de 0.6 en moyenne. Les résultats obtenus sont meilleurs lorsque les phrases de l'expert sont triées par longueur. On observe une augmentation du taux de réussite jusqu'à un palier (0.85 environ), malgré une difficulté croissante. En fin d'expérience, la difficulté des jeux provoque une légère baisse du taux de réussite. Le critère de longueur des phrases est plus formel et engendre une difficulté croissante dans l'expérience qui favorise l'apprentissage.

Nous donnons pour comparaison en Figure 5 les résultats obtenus pour la mesure de réussite communicative dans une baseline où les informations sémantiques relatives aux paires de tokens ne sont pas prises en compte dans leur représentation vectorielle. Lorsque le corpus est trié par âge de l'enfant destinataire, la réussite stagne rapidement autour de 0.45. Les résultats sont encore une fois meilleurs lorsque le corpus est trié par longueur des énoncés. Mais lorsqu'on ne tient pas compte des informations sémantiques, il faut plus de temps à l'apprenant pour atteindre son palier maximal, celui-ci est moins élevé, et l'augmentation de la longueur des énoncés fait décroître la réussite de manière significative. L'information sémantique sémantique facilite donc l'apprentissage et rend le système plus robuste à l'augmentation de la taille des énoncés.

Pour comparaison également est donnée en Figure 6 l'évolution de la réussite communicative pour une baseline où la classification est faite au hasard.

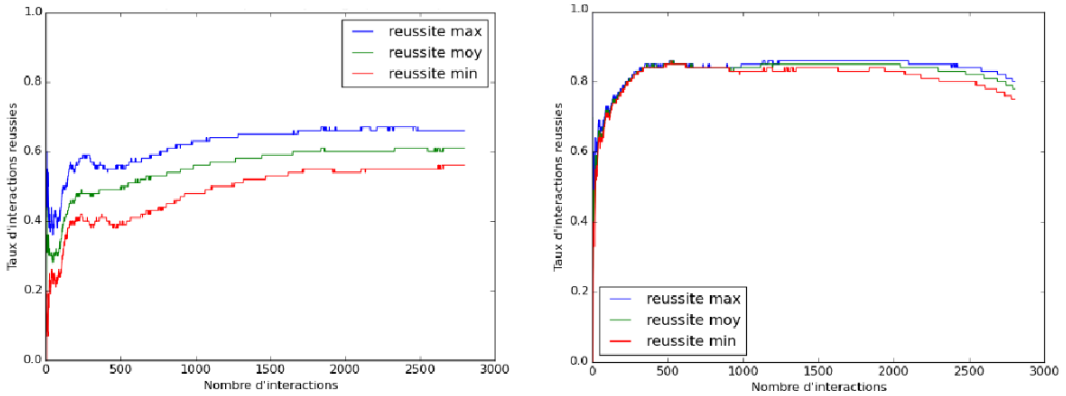


FIGURE 4 – Évolution de la proportion de jeux réussis en fonction du nombre d'interactions. À gauche, les énoncés sont triés par ordre croissant d'âge de l'enfant cible ; à droite, par longueur

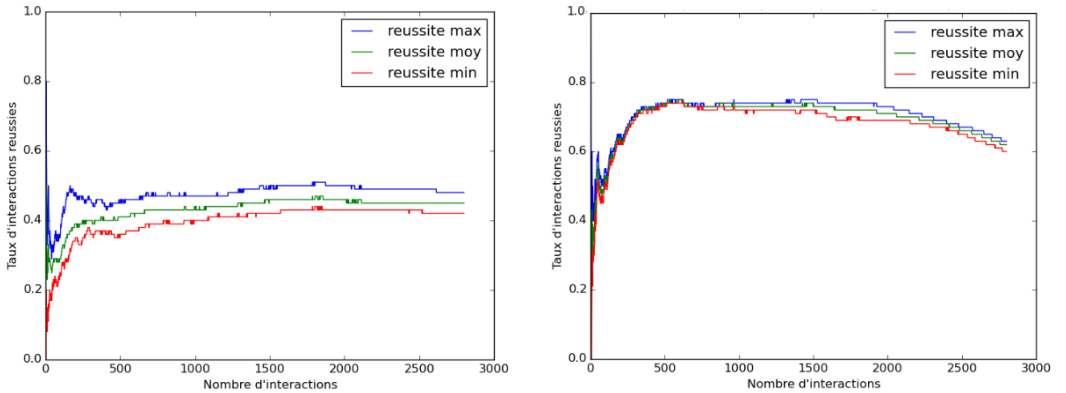


FIGURE 5 – Évolution de la proportion de jeux réussis en fonction du nombre d'interactions, avec une baseline sans informations sémantiques. À gauche, les énoncés sont triés par ordre croissant d'âge de l'enfant cible ; à droite, par longueur

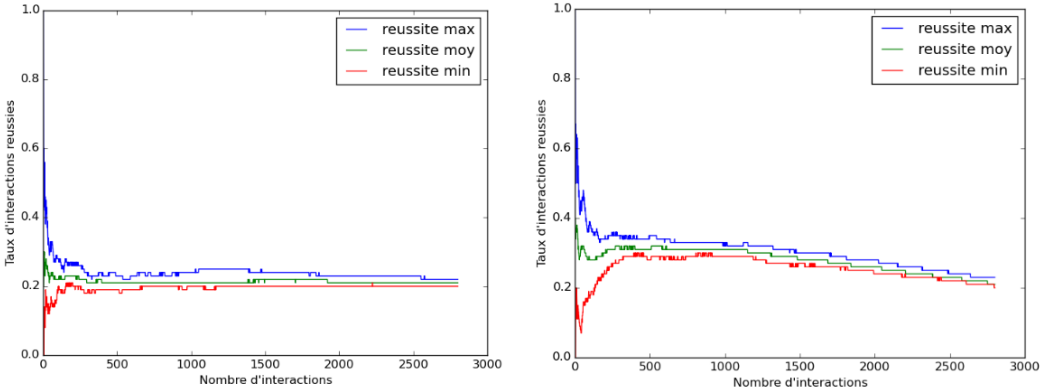


FIGURE 6 – Évolution de la proportion de jeux réussis en fonction du nombre d'interactions, avec une baseline où la classification est faite au hasard. À gauche, les énoncés sont triés par ordre croissant d'âge de l'enfant cible ; à droite, par longueur

4.2 Mesure UAS

La mesure d'UAS (Unlabeled Attachment Score) est plus classique pour évaluer la qualité d'un analyseur syntaxique relativement à un corpus de référence. Elle mesure la proportion de tokens pour lequel le bon gouverneur a été trouvé. Dans notre cas, nous ne disposons pas d'analyses de référence pour nos données, simplement du résultat fourni par l'application de MElt et Grew, qui peuvent faire des erreurs. Comme ce résultat sert de base à la sémantique, qui est utilisée par l'apprenant, c'est néanmoins par rapport à lui que nous nous évaluerons.

Le tableau 2 donne l'évolution de la mesure d'UAS au fil de l'expérience sur le corpus trié par longueur de phrases, avec (expérience standard) et sans utilisation de l'information sémantique (baseline 1), et lorsque la classification est faite au hasard (baseline 2), pour des portions de 400 phrases. On donne également la longueur moyenne des phrases pour chaque portion de corpus, et le nombre moyen de dépendances correctes trouvées (lors de l'expérience standard prenant en compte l'information sémantique).

Phrases	1-400	401-800	801-1200	1201-1600	1601-2000	2001-2400	2401-2800
UAS	0.87	0.78	0.67	0.69	0.61	0.58	0.46
UAS BL1	0.77	0.7	0.57	0.67	0.56	0.55	0.37
UAS BL2	0.33	0.31	0.3	0.29	0.28	0.26	0.26
Long. moy.	2.0	2.46	3.16	4.0	4.88	6.16	8.9
Nb. dép.	1.74	1.92	2.12	2.76	2.98	3.57	4.09

TABLE 2 – Mesure d'UAS (pour l'expérience standard, pour la baseline sans sémantique (BL1) et pour la baseline faisant une classification au hasard (BL2)), longueur moyenne des phrases et nombre moyen de dépendances correctes trouvées par phrase, par portion de 400 phrases du corpus au fil de l'expérience

Plus les phrases sont longues, plus l'UAS diminue (sauf entre les portions 801-1200 et 1201-1600). Malgré une difficulté croissante pour trouver le bon arbre de dépendances, l'apprenant réussit le jeu dans la proportion vue précédemment. On note également que, malgré une baisse de l'UAS, le nombre de dépendances correctes trouvées par l'apprenant augmente tout au long de l'expérience.

5 Conclusion et perspectives

Nous avons présenté un modèle d'apprentissage automatique de la syntaxe qui s'inspire de l'acquisition d'une langue naturelle par les enfants, et qui permet l'apprentissage d'un parser sans corpus arboré, mais avec seulement une représentation sémantique des phrases. Les résultats de ce modèle sont satisfaisants, mais peuvent être améliorés notamment par l'utilisation d'autres types de classifieurs (il est prévu d'implémenter un modèle utilisant des réseaux de neurones). Il doit pouvoir s'adapter à d'autres langues, ce qu'il faudra également tester.

Références

BASSANO D., LAAHA S., MAILLOCHON I. & DRESSLER W. U. (2004). Early acquisition of verb grammar and lexical development : Evidence from periphrastic constructions in french and austrian german. *First Language, 24(1)*, p. 33–70.

BICKERTON D. (1990). *Language and Species*. Chicago : The University of Chicago Press.

CHAMPAUD C. (1994). The development of verb forms in french children at around two years of age : some comparisons with romance and non-romance languages. *First Lisbon Meeting on Child Language*.

COPESTAKE A. (2009). Invited talk : Slacker semantics : Why superficiality, dependency and avoidance of commitment can be the right way to go. p. 1–9.

DEMUTH K. & TREMBLAY A. (2008). Prosodically-conditioned variability in children's production of french determiners. *Journal of Child Language, 35*, p. 99–127.

DENIS P. & SAGOT B. (2009). Coupling an annotated corpus and a morphosyntactic lexicon for state-of-the-art pos tagging with less human effort.

DESSALLES J.-L. (2000). *Aux origines du langage. Une histoire naturelle de la parole*. Paris : Hermes Science.

GOAD H. & BUCKLEY M. (2006). Prosodic structure in child french : Evidence for the foot. *Catalan Journal of Linguistics, 5*, p. 109–142.

GUILLAUME B. & PERRIER G. (2012). Annotation sémantique du french treebank à l'aide de la réécriture modulaire de graphes. *Actes TALN 2012*, p. 293–306.

HAMANN C., OHAYON S., DUBÉ S., FRAUENFELDER U., RIZZI L., STARKE M. & ZESIGER P. (2003). Aspects of grammatical development in young french children with sli. *Developmental Science, 6*, p. 151–159.

HARTLINE K., WAGNER H. & RATLIFF F. (1956). Inhibition in the eye of limulus. *Journal of General Physiology, 39*, p. 651–673.

HUNKELER H. (2005). Aspects of the evolution of the early lexicon in the interactions mother-child : Case study of two dizygotic twin children between 15 and 26 months.

KAPLAN F. (2001). *La naissance d'une langue chez les robots*.

KERN S., DAVIS B. L. & ZINK I. (2009). From babbling to first words in four languages : Common trends, cross language and individual differences. *First Lisbon Meeting on Child Language*.

KIRBY S. M. (2002). Natural language from artificial life. *Artificial Life, 8*, p. 185–215.

LEROY M., MATHIOT E. & MORGENSTERN A. (2009). Pointing gestures and demonstrative words : Deixis between the ages of one and three. p. 386–404.

MACWHINNEY B. (2000). *The CHILDES Project : Tools for analyzing talk. Third Edition*. Mahwah, NJ : Lawrence Erlbaum.

NORMAND M. T. L., MORENO-TORRES I., PARISSE C. & DELLATOLAS G. (2013). How do children acquire early grammar and build multiword utterances ? a corpus study of french children aged 2 to 4. *Child Development, 84(2)*, p. 647–661.

PALASIS K. (2010). *Syntaxe générative et acquisition : le sujet dans le développement du système linguistique du jeune enfant*.

PLUNKETT B. (2003). Null subjects and the setting of subject agreement parameters in child french. *Romance Linguistics : Theory and Acquisition*, p. 351–366.

P.SUPPES, SMITH R. & LEVEILLÉ M. (1973). The french syntax of a child's noun phrases. *Archives de Psychologie, 42*, p. 207–269.

STEELS L. (1995). A self-organizing spatial vocabulary. *Artificial Life*, p. 319–332.

STEELS L. (2000). Language as a complex adaptive system. p. 17–28.

STEELS L. & GARCIA-CASADEMONT E. (2015). How to play the syntax game. *Proceedings of the European Conference on Artificial Life 2015*, p. 479–486.

STEELS L. & KAPLAN F. (1999). Collective learning and semiotic dynamics. *Advances in Artificial Life. Proceedings of the Fifth European Conference, ECAL'99*, p. 679–688.

TELLIER I. (2005). Modéliser l'acquisition de la syntaxe du langage via l'hypothèse de la primauté du sens. HDR d'informatique, université Lille3.

VIHMAN M. M., DEPAOLIS R. A. & DAVIS B. L. (1998). Is there a "trochaic bias" in early word learning ? evidence from english and french. *Child Development, 69*, p. 933–947.

WELLENS P. (2012). *Adaptive Strategies in the Emergence of Lexical Systems*. Vrije Universiteit Brussel, Bruxelles : VUBPRESS Brussels University Press.

WINOGRAD T. (1971). *Procedures as a Representation for Data in a Computer Program for Understanding Natural Language*. Cambrige : Massachusetts Institute of Technology Project MAC Report.

YAMADA H. & MATSUMOTO Y. (2003). Statistical dependency analysis with support vector machines. *Proceedings of IWPT, vol. 3*, p. 195–206.

YAMAGUCHI N. (2012). *Parcours d'acquisition des sons du langage chez deux enfants francophones*. Université Sorbonne Nouvelle Paris 3.

MOTS : un outil modulaire pour le résumé automatique

Valentin Nyzam[1] Christophe Rodrigues[2] Aurélien Bossard[1]
(1) Laboratoire d'Informatique Avancée de Saint-Denis, Université Paris 8
2 rue de la Liberté, 93526 Saint-Denis, France
(2) De Vinci Research Center, ESILV
12 avenue Léonard de Vinci, 92400 Courbevoie, France
v.nyzam@iut.univ-paris8.fr, christophe.rodrigues@devinci.fr,
a.bossard@iut.univ-paris8.fr

RÉSUMÉ
Cet article présente un système open source et modulaire pour le résumé automatique : MOTS, développé en Java. Son architecture permet d'implémenter et tester de nouvelles méthodes de résumé automatique et de les comparer avec des méthodes existantes dans un cadre unifié. Ce système, le premier complètement modulaire pour le résumé automatique permet à l'heure actuelle de définir plus de cent combinaisons de modules afin de résumer automatiquement des textes en langage naturel.

ABSTRACT
MOTS : A Modular Framework for Automatic Summarization

This paper presents an open source and modular system for automatic summarization (AS) : MOTS, written in Java. Its architecture allows to implement and test new methods and to ease comparison with already implemented methods in an unified framework. It is the first completely modular system for AS and allows to summarize texts written in natural language using more than a hundred combinations of modules.

1 Introduction

L'évaluation de l'apport de différentes méthodes de résumé automatique, une discipline étudiée depuis les années 1950 (Luhn, 1958), est compliquée. En effet, les techniques de résumé automatique, quand elles sont rendues publiques, sont souvent incluses dans un système plus large qui comprend des pré et post-traitements ainsi que des ressources externes. Ceux-ci influent énormément sur la qualité des résumés produits. Il est donc important d'évaluer les techniques de résumé automatique dans un cadre commun afin de pouvoir les comparer précisément et juger de leur efficacité sur différents types de données.

Dans cet article, nous proposons une solution à ce problème : un outil de résumé automatique complètement modulaire et open source développé en Java qui permet de brancher ou débrancher les pré et post traitements. Cet outil permet également de bénéficier rapidement de *baselines* afin d'accélérer et de faciliter la recherche dans le domaine du résumé automatique.

Un tel outil est une réelle nouveauté. En effet, parmi les systèmes disponibles, on peut distinguer deux catégories : les systèmes issus d'une recherche originale et publiés par leurs auteurs comme MEAD [1]

1. http://www.summarization.com/mead/

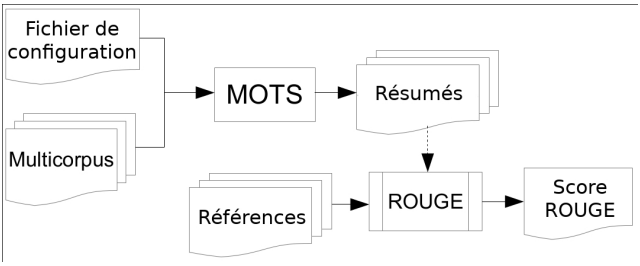


FIGURE 1 – Utilisation du système MOTS

(Radev *et al.*, 2004a), ICSISUMM [2] (Gillick *et al.*, 2009) ou encore MUSEEC [3] (Litvak *et al.*, 2016), qui n'implémentent que la ou les méthodes des auteurs et comprennent des pré et post-traitements spécifiques ; et les systèmes qui ré-implémentent plusieurs méthodes de résumé automatique, comme Sumy [4] et PKUSumSum [5] (Zhang *et al.*, 2016), qui en implémentent respectivement sept et dix. Cependant, les systèmes, y compris dans cette dernière catégorie, ont une modularité limitée au choix de la méthode de résumé automatique.

2 MOTS : Outil modulaire pour le résumé automatique

Pour se démarquer des systèmes cités en Section 1, afin de proposer une vraie solution à l'évaluation de l'apport de différents pré et post-traitements et pour permettre de l'adapter rapidement à une tâche spécifique, notre système de résumé automatique est complètement modulaire sur la totalité de la chaîne de traitement. Chacun de ses composants appartient à une classe spécifique qui code pour un rôle spécifique. Malgré son architecture complexe, MOTS reste facile à utiliser pour l'utilisateur final ; il ne nécessite que deux entrées : un corpus et un fichier de configuration (cf figure 1). MOTS est le plus générique possible afin qu'il puisse gérer ou être adapté à n'importe quel type de résumé automatique : extractif, semi-extractif [6] ou même entièrement abstractif.

2.1 Chaîne de traitement

L'architecture de MOTS est présentée en figure 2. Un multicorpus est ainsi défini comme un ensemble de corpus qui peuvent être résumés indépendemment les uns des autres. Un corpus peut être composé de un ou plusieurs textes, donc MOTS gère les résumés mono et multidocument. Dans cette même figure, un "modèle de résumé" est composé de trois étapes : pré-traitements, méthode de résumé et post-traitements. Une méthode de résumé est, elle, décomposée en quatre étapes qui suivent un schéma classique de résumé extractif ou semi-extractif. Premièrement, la méthode de résumé doit définir comment les "tokens" sont identifiés et indexés par le système. Ensuite, la méthode définit la représentation des phrases (*sentence characteristics*). Puis elle peut évaluer les phrases (toutes

2. https://code.google.com/archive/p/icsisumm
3. https://bitbucket.org/elenach/onr_gui/wiki/Home
4. https://github.com/miso-belica/sumy
5. https://github.com/PKULCWM/PKUSUMSUM/
6. résumé automatique semi-extractif : combinaison d'extraction et de traitements comme la compression ou la paraphrase.

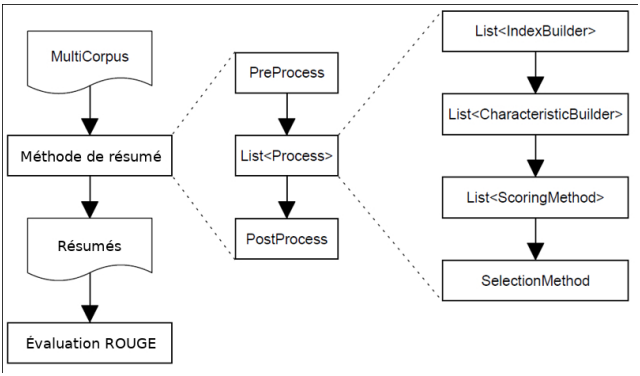


FIGURE 2 – Chaîne de traitement modulaire

les méthodes extractives ne requièrent pas d'évaluer les phrases). C'est l'étape "*sentence scoring*". Finalement, les phrases sont sélectionnées pour apparaître dans le résumé (*selection method*). Une méthode peut avoir de multiples constructeurs d'index, de caractéristiques et de multiples méthodes de *scoring* afin de combiner différentes approches. Les résumés produits par un modèle de résumé peuvent être évalués en utilisant ROUGE (Lin, 2004) qui est encapsulé dans notre système.

2.2 Modularité et implémentation pour MOTS

L'implémentation d'un nouveau modèle de résumé dans MOTS se réalise soit seulement en définissant un nouveau fichier de configuration qui fait appel aux modules existants, soit en définissant au préalable de nouveaux modules. Ces modules, ou traitements atomiques sont indépendants les uns des autres et la communication entre eux est gérée par la classe "Process" qui contrôle leur exécution et compatibilité. Les entrées et sorties des traitements atomiques sont spécifiées par héritage de méthodes et interfaces prédéfinies et décrites dans la documentation de MOTS, qui permettent à la classe "Process" d'utiliser des méthodes pour adapter les entrées et sorties de chaque traitement atomique. Les traitements atomiques sont indépendants les uns des autres et suivent des règles de compatibilité ; cette organisation du code rend MOTS complètement modulaire.

Toute personne désirant implémenter un nouveau traitement pour MOTS peut donc intervenir à n'importe quelle étape de la chaîne de traitement en suivant les règles définies par les interfaces idoines. De nombreux traitements de base utilisés par la majorité des systèmes de résumé automatique ont déjà été implémentés ; l'implémentation de nouvelles méthodes de résumé automatique en est facilitée.

L'outil MOTS est ouvert à tous les contributeurs sous licence GPL.

2.3 Algorithme génétique d'optimisation d'hyperparamètres

Les méthodes de résumé automatique comportent des paramètres qui influent sur la qualité des résumés générés. Selon (Litvak *et al.*, 2010; Bossard & Rodrigues, 2011), un algorithme génétique

```
<CONFIG>
  <TASK ID="1">
    <OPTION NAME="DampingParameter">true</OPTION>
    <OPTION NAME="GraphThreshold">true</OPTION>
    <OPTION NAME="Lambda">false</OPTION>
  </TASK>
</CONFIG>
```

FIGURE 3 – Exemple de fichier de configuration de l'algorithme génétique pour la méthode LexRank : seuls le *daping factor*, lambda et le seuil de similarité sont optimisés.

```
<CONFIG>
  <TASK ID="1">
    <LANGUalgorithme génétiqueE>english</LANGUalgorithme génétiqueE>
    <INPUT_PATH>doc</INPUT_PATH>
    <OUTPUT_PATH>doc/output</OUTPUT_PATH>
    <MULTITHREADING>true</MULTITHREADING>
    <PROCESS>
      <OPTION NAME="CorpusIdToSummarize">all</OPTION>
      <OPTION NAME="ReadStopWords">false</OPTION>
      <INDEX_BUILDER NAME="TF_IDF.TF_IDF">
      </INDEX_BUILDER>
      <CARACTERISTIC_BUILDER NAME="vector.query.Centroid">
        <OPTION NAME="NbMaxWordInCentroid">50</OPTION>
      </CARACTERISTIC_BUILDER>
      <SCORING_METHOD NAME="QuerySimilarity">
        <OPTION NAME="SimilarityMethod">
          JaccardSimilarity
        </OPTION>
      </SCORING_METHOD>
      <SUMMARIZE_METHOD NAME="MMR">
        <OPTION NAME="CharLimitBoolean">true</OPTION>
        <OPTION NAME="Size">100</OPTION>
        <OPTION NAME="Lambda">0.7</OPTION>
        <OPTION NAME="SimilarityMethod">
          JaccardSimilarity
        </OPTION>
      </SUMMARIZE_METHOD>
    </PROCESS>
    <ROUGE_EVALUATION>
      <ROUGE-MEASURE>
        ROUGE-1 ROUGE-2 ROUGE-SU4
      </ROUGE-MEASURE>
      <ROUGE-PATH>
        lib/ROUGE-1.5.5/RELEASE-1.5.5
      </ROUGE-PATH>
      <MODEL-ROOT>models</MODEL-ROOT>
      <PEER-ROOT>systems</PEER-ROOT>
    </ROUGE_EVALUATION>
  </TASK>
</CONFIG>
```

FIGURE 4 – Exemple d'un fichier de configuration pour MOTS

peut améliorer les résultats d'une méthode de résumé automatique donnée afin de construire un système de résumé automatique plus performant ou plus spécialisé. MOTS intègre un algorithme génétique pour optimiser les paramètres des méthodes de résumé automatique. Nous avons défini une syntaxe d'*adn* qui est utilisée pour décrire quels sont les paramètres d'une méthode à optimiser. Ces paramètres, définis dans les traitements atomiques, peuvent être ou non optimisés par l'algorithme génétique selon un fichier de configuration en xml passé en entrée de MOTS dont un exemple est donné en figure 3.

3 Méthodes implémentées

3.1 Indexation

MOTS inclut des méthodes pour indexer selon des unigrammes ou n-grammes, selon leur fréquence ou leur *tf.idf* (Salton & Buckley, 1988).

Est également implémentée une méthode d'indexation fondée sur *LDA* (Blei *et al.*, 2003) : k *topics* sont identifiés par une phase d'analyse Dirichlet latente sur les documents à résumer. Chaque *token* se voit alors attribuer une probabilité de distribution sur ces k *topics*.

Query LDA étend *LDA* en construisant la distribution des topics pour les documents. Les documents peuvent alors servir de requête pour évaluer la pertinence d'un fragment textuel (plus un fragment aura une distribution de *topics* proche de celle des documents, plus il sera pertinent).

Word embeddings génère les plongements lexicaux pour chaque *token* selon la méthode de (Mikolov *et al.*, 2013). Les *tokens* sont alors représentés par un vecteur de nombres décimaux qui exprime leur sémantique.

3.2 Caractérisation

L'étape de caractérisation sert à obtenir une représentation de séquences de *tokens* à partir de la représentation des *tokens* eux-mêmes.

MOTS implémente des méthodes vectorielles pour les phrases, les documents et les corpus : sac de mots *tf.idf*, construction d'un *centroïde* comme décrit dans (Radev *et al.*, 2004b), ainsi que leur extension matricielle. MOTS implémente également des caractéristiques fondées sur les graphes : graphes de co-occurrence (Rousseau & Vazirgiannis, 2013), K-core (Batagelj & Zaversnik, 2003), ainsi que des méthodes de regroupement automatique de phrases fondées sur les caractéristiques obtenues par une autre des méthodes de caractérisation définies.

3.3 Notation de fragments textuels

La notation de fragments textuels utilise les représentations calculées au préalable par l'étape de caractérisation afin de noter les fragments en vue de leur éventuelle sélection dans le résumé.

Les méthodes de notation suivantes sont implémentées dans MOTS :

- *LexRank* (Erkan & Radev, 2004) se fonde sur la notion de popularité dans un graphe afin de détecter les fragments les plus centraux d'un ou plusieurs documents (Erkan & Radev, 2004) ;
- *tf.idf threshold* somme les *tf.idf* des tokens d'un fragment au dessus d'un seuil ;
- *QuerySimilarity* évalue un fragment de texte vis-à-vis d'une requête vectorielle construite dans l'étape précédente – cela permet par exemple de réaliser la méthode centroïde (Radev *et al.*, 2004b) dans MOTS.

3.4 Sélection

Méthodes incrémentales MOTS implémente trois méthodes de sélection incrémentale de phrases.

BestIsBetter extrait naïvement les meilleurs fragments selon une des notations définies en §3.3 et risque donc, surtout en contexte multidocument, de produire des résumés redondants.

MMR sélectionne à chaque itération la phrase qui est à la fois la plus centrale et la moins similaire aux phrases déjà sélectionnées selon la méthode de (Carbonell & Goldstein, 1998), donc le fragment qui maximise la formule suivante :

$$MMR(D,R) = \arg\max_{F_i \in D \setminus R} [\lambda \times centralite(F_i) - (1-\lambda) \max_{F_j \in R} (sim(F_i, F_j))]$$

où D est l'ensemble des documents à résumer et R l'ensemble des fragments déjà extraits dans le résumé, centralité une mesure de centralité et sim une mesure de similarité entre fragments.

CSIS extrait itérativement le meilleur fragment dont la similarité à un fragment déjà sélectionné n'excède pas un seuil prédéfini, comme décrit dans (Radev *et al.*, 2004a).

Méthodes d'exploration l'outil inclut également des méthodes d'exploration de l'espace des résumés candidats. Ces méthodes permettent de pallier le principal défaut des méthodes incrémentales : les résumés produits sont trop dépendants de la première phrase sélectionnée.

ILP optimise une fonction de score linéaire sous contraintes en nombres entiers (Gillick & Favre, 2009). La fonction à optimiser est une simple somme pondérée des *tokens* avec leur nombre d'occurrences comme poids. La méthode d'indexation définit donc quel *token* sera pris en compte dans la fonction objectif d'*ILP*. Cette méthode, avec la fonction objectif fondée sur les bigrammes, est reconnue à la fois pour sa rapidité d'exécution et son efficacité sur des tâches de résumé multidocument.

Knapsack utilise un algorithme dynamique de résolution du problème du sac à dos décrit dans (McDonald, 2007) afin d'optimiser une fonction objectif en prenant en compte la contrainte de la taille des résumés.

Genetic implémente la méthode décrite dans (Bossard & Rodrigues, 2017), qui optimise une fonction objectif grâce à un algorithme évolutionnaire. Cet algorithme évolutionnaire a l'avantage de optimiser n'importe quelle fonction objectif a un temps d'exécution conséquent.

Pour ces deux dernières méthodes, différentes fonctions objectif sont disponibles : similarité au document, à une requête, divergence distributionnelle... Tout comme pour l'architecture globale de MOTS et l'interface de notation de fragments textuels, il existe une interface générique dont doivent hériter toute nouvelle fonction objectif.

Sont également incluses des méthodes de résumé automatique par l'apprentissage par renforcement, qui doit explorer un espace de solution conséquent et montre les limites de ce type de méthodes appliquées au résumé automatique (Sutton & Barto, 1998; Ryang & Abekawa, 2012).

Méthodes par abstraction Les méthodes de résumé par abstraction connaissent un regain d'intérêt depuis 2015 (Rush *et al.*, 2015). Ce regain d'intérêt est principalement dû aux capacités génératives des réseaux de neurones récurrents, qui ont montré une efficacité certaine sur la traduction automatique. Ces dernières années ont vu beaucoup d'évolution dans les modèles d'apprentissage afin de les adapter au mieux au résumé automatique malgré la difficulté principale inhérente à ce domaine : un espace de recherche très étendu par rapport à la traduction automatique (il s'agit pour résumer de "traduire" un ou plusieurs documents en plusieurs phrases, et non une phrase en une autre).

Étant donné le renouveau des méthodes abstractives pour le résumé automatique, nous ne pouvions pas uniquement dédier MOTS au résumé extractif et semi-extractif. Nous y avons donc également implémenté un module qui permet d'appeler un système de résumé par abstraction et d'intégrer ses résultats à MOTS. Cette intégration n'est pas évidente : les systèmes par abstraction, notamment ceux à base de réseaux de neurones récurrents dont le modèle a été appris au préalable peuvent produire des mots en dehors du vocabulaire des documents d'origine. Ces mots doivent alors être intégrés a posteriori à l'index afin de pouvoir réaliser les post-traitements dans MOTS sur les résumés produits par ce module.

4 Utiliser MOTS

Le système MOTS est conçu pour être facile à utiliser et à y contribuer malgré son architecture relativement complexe. En tant qu'utilisateur final, seules sont nécessaires la mise au format du

corpus ainsi que l'écriture d'un fichier de configuration qui décrit le modèle de résumé qui servira à générer les résumés. Pour plus de simplicité, des fichiers de configuration sont prédéfinis dans le système. La figure 4 présente un fichier de configuration qui décrit un modèle qui utilise le score des phrases par la méthode "centroïde" et leur sélection par la méthode "MMR".

Ainsi, avec MOTS, l'installation d'un système unique et une mise au format du corpus suffisent pour récupérer les résumés produits par plusieurs méthodes état de l'art, contre l'installation de plusieurs systèmes et plusieurs conversions du corpus pour le même résultat sans MOTS. Les résultats différeront forcément légèrement des systèmes d'origine. En effet, les pré-traitements utilisés ne sont pas les mêmes, et les systèmes d'origine peuvent différer légèrement dans l'utilisation de procédures annexes non décrites dans les articles. Cependant, implémenter une nouvelle méthode de résumé au sein de MOTS permet de la comparer à des méthodes existantes dans exactement les mêmes conditions.

5 Évaluation

5.1 Protocole

Nous décrivons ici le protocole d'évaluation de MOTS, qui permet d'évaluer les méthodes implémentées et de les comparer quand c'est possible aux systèmes d'origine qu'elles émulent. Les méthodes sont évaluées sur les corpus DUC 2006/2007, TAC 2008/2009/2010 sur leur partie standard et sans prise en compte des requêtes et dont les caractéristiques sont données dans le tableau 4.

5.2 Méthodes évaluées

Toutes les méthodes partagent les mêmes pré et post-traitements : StanfordNLP pour le découpage en mots/phrases (Manning *et al.*, 2014) et le *stemmer* de Porter pour la racinisation. Le tableau 1 détaille les processus atomiques utilisés dans les méthodes évaluées à des fins de reproductibilité.

JSBigram Knapsack utilise la méthode d'exploration *Knapsack* décrite dans (McDonald, 2007) pour optimiser une métrique d'évaluation de résumé automatique fondée sur la distance de Jensen-Shannon décrite dans (Louis & Nenkova, 2009).

JSBigram Genetic utilise la méthode d'optimisation évolutionnaire *Genetic* pour optimiser la même métrique que "JSBigram Knapsack" et émule donc la méthode décrite dans (Bossard & Rodrigues, 2017).

JSBigram Reinforcement utilise la méthode d'apprentissage par renforcement décrite dans (Ryang & Abekawa, 2012) pour optimiser la même métrique que les deux méthodes précédentes.

Bigram ILP émule la méthode de résumé automatique par programmation linéaire en nombres entiers de (Gillick & Favre, 2009).

LexRank MMR utilise la méthode de résumé automatique LexRank (Radev *et al.*, 2004a) comme score de phrases et MMR (Carbonell & Goldstein, 1998) pour les extraire dans le résumé.

LexRank MMR opt optimise sur le corpus TAC 2009 les trois paramètres *damping factor, epsilon et seuil de similarité* de la méthode "LexRank MMR" grâce à l'algorithme génétique de MOTS.

	Indexation	Caractéristiques	Score	Sélection
JSBigram Knapsack	Word, 2-gram	-	-	Knapsack (JS metric)
JSBigram Genetic	Word, 2-gram	-	-	Genetic (JS metric)
Bigram ILP	Word, 2-gram	-	-	ILP
LexRank MMR	Word	tf.idf vector	LexRank (Jaccard)	MMR (cosine)
TfIdf MMR Cosine	Word	tf.idf document	query sim (cosine)	MMR (cosine)
LDA	Query LDA	mean vector sent	query sim (JS)	BestIsBetter
Bigram Centr. MMR	Word, 2-gram	centroid	query sim (Jaccard)	MMR (Jaccard)
KCore Query	Word	KCore query	query sim (Jaccard)	MMR (Jaccard)
JSBigram Reinforc.	Word, 2-gram	tf.idf vector	-	Reinforc.Learn. (JS)
W2V LexRank MMR	Word embeddings	mean vector sent	LexRank (cosine)	MMR (cosine)

TABLE 1 – Résumé des processus atomiques utilisés dans les méthodes évaluées

TF-IDF MMR Cosinus émule la méthode du centroïde (Radev *et al.*, 2004b) pour évaluer les phrases, et MMR pour les sélectionner.

LDA utilise l'analyse Dirichlet latente pour représenter les phrases et le corpus comme des distributions de topics (Blei *et al.*, 2003). La distribution du document est utilisée comme requête.

Bigram Centroid MMR est la même méthode que centroïde (Radev *et al.*, 2004b) mais utilise les bigrammes et non les unigrammes comme tokens.

KCore Query consiste à construire un graphe de co-occurrences des tokens comme dans (Rousseau & Vazirgiannis, 2013). La dégénérescence du graphe est calculée pour obtenir une décomposition en K-core (Batagelj & Zaversnik, 2003). Le meilleur K-core est utilisé comme requête et MMR comme méthode d'extraction.

W2V LexRank MMR utilise le vecteur moyen des plongements lexicaux d'une phrase comme caractéristique, LexRank/MMR pour extraire les phrases.

5.3 Baselines

Nous incluons les résultats obtenus par quatre systèmes externes publiés par leurs auteurs : **ICSISUMM query/ICSISUMM wo query** [7] décrit dans Gillick *et al.* (2009) et comparable à la méthode "Bigram ILP". "ICSISUMM wo query" débranche un filtre dans ICSISUMM lié à la proximité avec les requêtes des corpus DUC/TAC non utilisées dans notre système.

Genetic official : le résumé automatique par algorithme évolutionnaire (Bossard & Rodrigues, 2017) comparable à notre méthode "JSBigram Genetic".

MEAD [8] avec l'algorithme "centroïde", comparable à notre méthode "Centroid MMR".

5.4 Résultats

L'évaluation réalisée avec l'outil ROUGE et les paramètres que Graham (2015) a trouvé les plus corrélés aux scores manuels [9] est présentée en tableau 2. Ces résultats semblent cohérents avec les

7. https://github.com/benob/icsisumm
8. http://www.summarization.com/mead/
9. Les paramètres exacts sont : -n 2 -x -m -c 95 -r 1000 -f A -p 1 -t 0 -a -s

Corpora	D2006	D2007	T2008	T2009	T2010
ICSISUMM query	.07617	**.09952**	**.10625**	**.10108**	.08343
ICSISUMM wo query	.07286	.09753	.10286	.09150	.08973
Genetic official	**.07678**	.08591	.10448	.09537	**.09892**
MEAD	.05149	.06323	.05075	.04561	.06427
JSBigram Knapsack	.06882	**.09015**	**.10923**	.08732	.08853
JSBigram Genetic	**.07357**	.08399	.10544	.08576	.08900
Bigram Centr. MMR	.07328	.08276	.10658	.08112	.08848
Bigram ILP	.06977	.08077	.10123	.08251	**.09152**
LexRank MMR opt	.06952	.07805	.09150	**.09369**	.07714
LexRank MMR	.07094	.07917	.09370	.08609	.08128
JSBigram Reinforc.	.06779	.07160	.09637	.07310	.07328
KCore Query MMR	.05908	.07020	.08659	.06665	.07721
TFIDF MMR Cosine	.05786	.07291	.07990	.06383	.07110
LDA	.05659	.07113	.07471	.07275	.06919
W2V LexRank MMR	.06504	.06723	.04501	.04564	.05201

TABLE 2 – Scores ROUGE-2 sur tous les corpus

Corpora	D2006	D2007	T2008	T2009	T2010
ICSISUMM	651s	337s	95s	85s	56s
Genetic official	7121s	5123s	2047s	1951s	1911s
MEAD	186s	86s	74s	85s	56s
JSBigram Knapsack	1941s	1076s	281s	250s	217s
JSBigram Genetic	7950s	5850s	2492s	2375s	2251s
Bigram ILP	338s	181s	48s	42s	60s
Bigram Centr. Jacc MMR	74s	33s	12s	11s	10s
LexRank MMR Jacc	37s	20s	7s	8s	7s
KCore Query MMR Jacc	21s	12s	6s	6s	7s
JSBigram Reinforc.	1041s	672s	286s	266s	263s
TFIDF MMR Cosine	18s	12s	5s	6s	6s
LDA	58s	41s	21s	21s	20s
W2V LexRank MMR	134s	131s	104s	101s	103s

TABLE 3 – Temps d'exécution sur tous les corpus

méthodes état de l'art testées, même si de légères différences dans les résultats semblent valider l'importance des pré et post-traitements dans la qualité des résumés produits.

La méthode de résumé automatique fondée sur les plongements lexicaux n'est pas optimale et est peu performante. La comparaison de "ICSISUMM query" et "ICSISUMM wo query" montre l'importance de pré-traitements adaptés à la tâche : "ICSISUMM query" prend en compte les requêtes de chaque *topic* tandis que la prise en compte des requêtes est désactivée dans "ICSISUMM wo query". Nos meilleurs modèles de résumé automatique sont en effet meilleurs en moyenne que "ICSISUMM wo query" mais sont dépassés par "ICSISUMM query".

L'algorithme génétique embarqué par MOTS a été utilisé pour optimiser la méthode 'LexRank MMR opt" sur TAC2009. Les scores sur ce corpus sont nettement améliorés bien que l'amélioration soit limitée à ce corpus. Les hyperparamètres que nous avons optimisés sont uniquement ceux de l'algorithme de sélection, et ne sont sans doute pas assez généralisables. Pour obtenir une optimisation par l'algorithme génétique plus convaincante, il serait sans doute préférable d'inclure des hyperparamètres

liés au traitement des données en entrée, comme des poids différents pour chaque catégorie de *tokens* (entités nommées, verbes, adverbes...).

	DUC2006	DUC2007	TAC2008	TAC2009	TAC2010
topics	50	45	48	44	46
words/topic	17728	13693	6234	6679	5790

TABLE 4 – Nombre de *topics* et de mots par *topic* dans chaque corpus

Le Tableau 3 montre le temps d'exécution de chaque modèle sur chaque corpus. Les tâches DUC et TAC diffèrent das le nombre de mots à produire par résumé et dans le nombre de mots en entrée (cf tableau 4). Les comparer donne donc une idée de possibilités de passage à l'échelle des modèles de résumé automatique.

Notre évaluation a mis en valeur une nouvelle méthode qui dépasse les systèmes fondées sur les algorithme génétique et ILP : "JSBigram Knapsack". Cela valide l'intérêt de la modularité de notre système, qui a permis de tester facilement différentes combinaisons de modules, dont celle-ci.

5.5 Discussion

Les résultats mettent en évidence des différences de scores assez importantes, notamment en ce qui concerne les systèmes "MEAD" et "Bigram Centroid MMR". Ce dernier se révèle d'une fois et demie à deux fois plus performant en termes de scores ROUGE que MEAD. MEAD par défaut une combinaison de scores, dont "Centroïde" et la position des phrases dans le document. Ce dernier score est sensé améliorer les résultats de "Centroïde" seul, il est donc d'autant plus étonnant de constater de telles différences de score ROUGE.

Nous ne pouvons les expliquer que par les pré-traitements des deux systèmes. MOTS utilise par défaut StanfordNLP pour le découpage en mots et en phrases ainsi que le *stemmer* de Porter pour raciniser les *tokens*, tandis que "MEAD" utilise les mots pleins. Ces différences de score ne font qu'appuyer notre hypothèse initiale, à savoir que les pré et post-traitements ont une influence importante sur les résultats d'un système de résumé automatique.

6 Conclusion et perspectives

Cet article présente MOTS, le premier système complètement modulaire pour le résumé automatique. Ce système est *open source* et librement disponible. Il vise à faciliter la comparaison dans un cadre unifié de nouvelles méthodes de résumé automatique avec des méthodes existantes et implémentées dans MOTS. MOTS est disponible sur GitHub [10], pour que chacun puisse y contribuer. MOTS contient un algorithme génétique qui optimise les hyperparamètres des modèles de résumé automatique.

La comparaison des résultats des modèles de résumé automatique de MOTS avec les résultats des systèmes d'origine qu'ils émulent montrent le besoin d'évaluations différentes qui permettent de quantifier l'apport réel de méthodes de résumé en les isolant des pré et post-traitements.

10. Nom sur github changé pour préserver l'anonymat

Même si nous n'avons évalué que dix méthodes de résumé automatique, la modularité de MOTS permet, en combinant les processus atomiques, de définir plus d'une centaine de méthodes de résumé automatique. Nous avons également présenté une méthode de résumé automatique inédite fondée sur la décomposition en K-Cores avec des premiers résultats mitigés. La modularité de MOTS nous a permis de tester aisément différentes combinaisons de processus atomiques et de mettre en valeur une nouvelle méthode de résumé automatique, rapide et efficace, qui utilise un algorithme de programmation dynamique de résolution du problème du sac à dos guidé par la divergence de Jensen-Shannon.

Nous avons implémenté un module qui permet d'appeler un système de résumé abstractif externe, récupérer ses résultats et les intégrer à MOTS pour d'éventuels post-traitements. Ce module permet ainsi d'utiliser les avancées récentes dans le domaine des réseaux de neurones profonds et leur application au résumé automatique (Hua & Wang, 2017; Tan *et al.*, 2017; Zhou *et al.*, 2017; Chopra *et al.*, 2016).

Le système MOTS est ouvert à tous les contributeurs. Nous implémentons actuellement d'autres méthodes état de l'art (p.e. les fonctions submodulaires) et améliorons certaines caractéristiques (p.e. les plongements lexicaux). Nous espérons que la communauté du TAL trouvera MOTS utile et qu'il gagnera de nouveaux contributeurs.

Remerciements

Ce travail a bénéficié d'une aide de l'Agence Nationale de la Recherche portant la référence ANR-16-CE38-0008 (projet ANR JCJC ASADERA).

Références

BATAGELJ V. & ZAVERSNIK M. (2003). An o(m) algorithm for cores decomposition of networks. *CoRR*, **cs.DS/0310049**.

BLEI D. M., NG A. Y. & JORDAN M. I. (2003). Latent dirichlet allocation. *Journal of machine Learning research*, **3**(Jan), 993–1022.

BOSSARD A. & RODRIGUES C. (2011). Combining a multi-document update summarization system–cbseas–with a genetic algorithm. In *Combinations of intelligent methods and applications*, p. 71–87. Springer.

BOSSARD A. & RODRIGUES C. (2017). An evolutionary algorithm for automatic summarization. In *Proceedings of the International Conference Recent Advances in Natural Language Processing, RANLP 2017*, p. 111–120, Varna, Bulgaria : INCOMA Ltd.

CARBONELL J. & GOLDSTEIN J. (1998). The use of mmr, diversity-based reranking for reordering documents and producing summaries. In *Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval*, p. 335–336 : ACM.

CHOPRA S., AULI M. & RUSH A. M. (2016). Abstractive sentence summarization with attentive recurrent neural networks. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics : Human Language Technologies*, p. 93–98, San Diego, California : Association for Computational Linguistics.

ERKAN G. & RADEV D. R. (2004). Lexrank : Graph-based lexical centrality as salience in text summarization. *Journal of AIR*, **22**, 457–479.

GILLICK D. & FAVRE B. (2009). A scalable global model for summarization. In *Proceedings of the Workshop on Integer Linear Programming for Natural Langauge Processing*, p. 10–18 : Association for Computational Linguistics.

GILLICK D., FAVRE B., HAKKANI-TÜR D., BOHNET B., LIU Y. & XIE S. (2009). The icsi/utd summarization system at tac 2009. In *Proc. of the Text Analysis Conference workshop, Gaithersburg, MD (USA)*.

GRAHAM Y. (2015). Re-evaluating automatic summarization with bleu and 192 shades of rouge. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, p. 128–137, Lisbon, Portugal : Association for Computational Linguistics.

HUA X. & WANG L. (2017). A pilot study of domain adaptation effect for neural abstractive summarization. In *Proceedings of the EMNLP Workshop on New Frontiers in Summarization*, Copenhagen, Denmark : Association for Computational Linguistics.

LIN C.-Y. (2004). Rouge : A package for automatic evaluation of summaries. In *Text summarization branches out : Proceedings of the ACL-04 workshop*, volume 8 : Barcelona, Spain.

LITVAK M., LAST M. & FRIEDMAN M. (2010). A new approach to improving multilingual summarization using a genetic algorithm. In *Proceedings of the 48th annual meeting of the association for computational linguistics*, p. 927–936 : Association for Computational Linguistics.

LITVAK M., VANETIK N., LAST M. & CHURKIN E. (2016). Museec : A multilingual text summarization tool. In *Proceedings of ACL-2016 System Demonstrations*, p. 73–78 : Association for Computational Linguistics.

LOUIS A. & NENKOVA A. (2009). Automatically evaluating content selection in summarization without human models. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing : Volume 1-Volume 1*, p. 306–314 : Association for Computational Linguistics.

LUHN H. P. (1958). The automatic creation of literature abstracts. *IBM J. Res. Dev.*, **2**(2), 159–165.

MANNING C. D., SURDEANU M., BAUER J., FINKEL J., BETHARD S. J. & MCCLOSKY D. (2014). The Stanford CoreNLP natural language processing toolkit. In *Association for Computational Linguistics (ACL) System Demonstrations*, p. 55–60.

MCDONALD R. (2007). A study of global inference algorithms in multi-document summarization. *Advances in Information Retrieval*, p. 557–564.

MIKOLOV T., SUTSKEVER I., CHEN K., CORRADO G. S. & DEAN J. (2013). Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, p. 3111–3119.

RADEV D. R., ALLISON T., BLAIR-GOLDENSOHN S., BLITZER J., CELEBI A., DIMITROV S., DRABEK E., HAKIM A., LAM W., LIU D. *et al.* (2004a). Mead-a platform for multidocument multilingual text summarization. In *LREC*.

RADEV D. R., JING H., STYŚ M. & TAM D. (2004b). Centroid-based summarization of multiple documents. *Information Processing & Management*, **40**, 919–938.

ROUSSEAU F. & VAZIRGIANNIS M. (2013). Graph-of-word and tw-idf : new approach to ad hoc ir. In *Proceedings of the 22nd ACM international conference on Information & Knowledge Management*, p. 59–68 : ACM.

RUSH A. M., CHOPRA S. & WESTON J. (2015). A neural attention model for abstractive sentence summarization. In *EMNLP*, p. 379–389.

RYANG S. & ABEKAWA T. (2012). Framework of automatic text summarization using reinforcement learning. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, p. 256–265 : Association for Computational Linguistics.

SALTON G. & BUCKLEY C. (1988). Term-weighting approaches in automatic text retrieval. *Information processing & management*, **24**(5), 513–523.

SUTTON R. S. & BARTO A. G. (1998). *Reinforcement learning : An introduction*, volume 1. MIT press Cambridge.

TAN J., WAN X. & XIAO J. (2017). Abstractive document summarization with a graph-based attentional neural model. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1 : Long Papers)*, p. 1171–1181 : Association for Computational Linguistics.

ZHANG J., WANG T. & WAN X. (2016). PKUSUMSUM : A java platform for multilingual document summarization. In *COLING (Demos)*, p. 287–291 : ACL.

ZHOU Q., YANG N., WEI F. & ZHOU M. (2017). Selective encoding for abstractive sentence summarization. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1 : Long Papers)*, p. 1095–1104 : Association for Computational Linguistics.

Ordonnancement de réponses dans les systèmes de dialogue basé sur une similarité contexte/réponse

Basma El Amel Boussaha Nicolas Hernandez Christine Jacquin
Emmanuel Morin
Laboratoire des Sciences du Numérique de Nantes (LS2N UMR 6004)
2 rue de la houssinière, BP 92208, 44322 Cedex 3 Nantes, France
prénom.nom@ls2n.fr

RÉSUMÉ

Construire des systèmes de dialogue qui conversent avec les humains afin de les aider dans leurs tâches quotidiennes est devenu une priorité. Certains de ces systèmes produisent des dialogues en cherchant le meilleur énoncé (réponse) parmi un ensemble d'énoncés candidats. Le choix de la réponse est conditionné par l'historique de la conversation appelé contexte. Ces systèmes ordonnent les énoncés candidats par leur adéquation au contexte, le meilleur est ensuite choisi. Les approches existantes à base de réseaux de neurones profonds sont performantes pour cette tâche. Dans cet article, nous améliorons une approche état de l'art à base d'un dual encodeur LSTM. En se basant sur la similarité sémantique entre le contexte et la réponse, notre approche apprend à mieux distinguer les bonnes réponses des mauvaises. Les résultats expérimentaux sur un large corpus de chats d'Ubuntu montrent une amélioration significative de 7, 6 et 2 points sur le Rappel@(1, 2 et 5) respectivement par rapport au meilleur système état de l'art.

ABSTRACT

Response ranking in dialogue systems based on context-response similarity

Building dialogue systems that converse with humans in order to help them in their daily tasks is becoming a priority. Some of these systems produce dialogues by finding the best response among a set of candidate responses. The choice of the best response is based on the history of the conversation called context. These systems rank the candidate responses by their relevance to the context, the best response is then chosen. Approaches based on deep neural networks performed well on this task. In this work, we improve a state of the art approach based on an LSTM dual encoder. By capturing semantic similarities between the context and the response, our approach learns to match the context with the best response. Experimental results on the Ubuntu Dialogue Corpus have shown a significant improvement of about 7, 6 and 2 points on Recall@(1, 2 and 5) compared to the best state of the art system.

MOTS-CLÉS : conversations écrites, dual encodeur, ordonnancement, agents conversationnels, apprentissage profond.

KEYWORDS: written conversations, dual encoder, ranking, chatbots, deep learning.

1 Introduction

Face au nombre croissant d'internautes, l'assistance automatique demeure la solution la plus adaptée pour les aider à résoudre leurs problèmes quotidiens. Grâce aux systèmes conversationnels, une assistance automatique peut être garantie pour chacun d'eux avec un coût minimal [1]. Ces systèmes appelés *chatbots* (agents conversationnels) sont capables de comprendre les besoins de l'utilisateur à travers des échanges textuels pour ensuite lui proposer une solution. Selon la nature de ces systèmes conversationnels, nous distinguons deux types (Figure 1) : les systèmes génératifs et les systèmes d'ordonnancement de réponses (Lowe *et al.*, 2017b). Les premiers systèmes génèrent des énoncés mot par mot, quant aux seconds, ils sélectionnent le bon énoncé parmi un ensemble d'énoncés candidats. En outre, et selon la tâche, nous distinguons deux catégories de systèmes conversationnels. La première catégorie regroupe les systèmes spécifiques à un domaine comme les systèmes de recommandation des restaurants (Wen *et al.*, 2017) et de réservation des tickets de cinéma (Li *et al.*, 2017). La deuxième catégorie comprend les systèmes de dialogue non spécifiques au domaine tels que SIRI [2], Alexa [3] et Replika [4]. Dans ce travail nous étudions les systèmes conversationnels qui sont à la fois des systèmes d'ordonnancement de réponses et spécifiques au domaine.

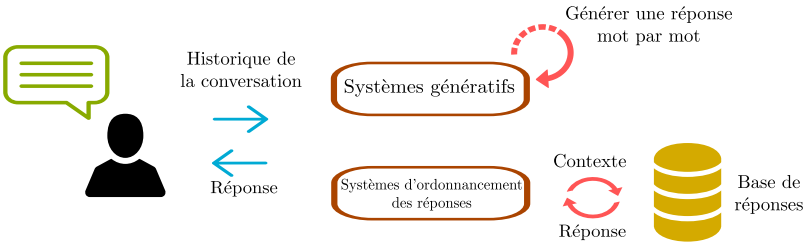


FIGURE 1: Les types des systèmes de dialogue.

Les conversations contiennent plusieurs tours de paroles entre deux ou plusieurs individus. Afin de produire une réponse adéquate à une conversation, il est important de considérer tous ces tours (nature multi-tours) ce qui rend la tâche plus complexe. Plusieurs travaux ont abordé le problème de la sélection du prochain tour de parole dans les conversations écrites. Certains exploitent tous les tours de parole de ces conversations pour sélectionner la réponse la plus adéquate à chacun d'eux (Lowe *et al.*, 2015; Kadlec *et al.*, 2015; Xu *et al.*, 2017; Wu *et al.*, 2017). D'autres négligent cette information pour ordonner les réponses candidates selon leur pertinence vis-à-vis du dernier tour de parole (Wang *et al.*, 2013; Wu *et al.*, 2016).

Les systèmes d'ordonnancement de réponses classent un ensemble de réponses candidates en se basant sur leur cohérence par rapport au contexte de la conversation. Dans la table 1, un exemple d'une conversation technique entre deux internautes extraite du corpus de dialogue d'Ubuntu (Lowe *et al.*, 2015) est illustré. Nous appelons *contexte*, l'historique de la conversation dans lequel nous concaténons les tours de parole de ces deux internautes. Dans cet exemple, un système d'ordonnancement de réponses doit classer la première réponse avant la deuxième. Il est important que le système capture les informations en commun (portées par les mots en gras) entre le contexte et chacune des réponses candidates. Selon Wu *et al.* (2017), les difficultés de la tâche d'ordonnancement de réponses

1. Comparé aux coûts engendrés par les humains travaillant comme assistants.
2. `https://www.apple.com/ios/siri/`
3. `https://developer.amazon.com/alexa`
4. `http://replika.ai/`

proviennent d'une part, de comment identifier les informations importantes (mots, phrases et énoncés) dans le contexte et de comment apparier ces informations avec celles présentes dans la réponse et d'autre part, de comment modéliser les relations entre les énoncés du contexte.

	Contexte
Tour 1	Hi, I can not longer access the graphical **login screen** on ubuntu 12.04
Tour 2	What exactly happen ?
Tour 3	I can't remember the error message, would it have auto-logged to a file or should I reboot quick ?
Tour 4	You mean it won't **automaticaly start** and what happen then ?
Tour 5	It just stop at a text **screen**, but I can access the command line **login** via alt F1-6, and **start** x **manually** there. I think it might me **lightdm** that's break but I'm not sure
	Réponses candidates
Réponse 1	For me **lightdm** often won't start **automatically** either. It show me console tty1 instead and I have to **start lightdm manually** ✓
Réponse 2	What about sources.list ? ✗

TABLE 1: Exemple d'une conversation technique entre deux participants extraite du corpus de dialogue d'Ubuntu (Lowe *et al.*, 2015). La première réponse candidate est la bonne réponse, tandis que la deuxième réponse ne peut être le prochain tour de parole.

Les travaux récents soit utilisent des architectures complexes pour capturer des similarités entre le contexte et la réponse ou nécessitent des modules externes afin de produire des informations complémentaires utiles à la tâche (Lowe *et al.*, 2015; Kadlec *et al.*, 2015; Wu *et al.*, 2016; Xu *et al.*, 2017). Certains de ces modules requièrent en plus, des connaissances externes collectées manuellement et qui sont fortement liées à un domaine d'application spécifique.

Dans ce travail, nous améliorons un système d'ordonnancement de réponses à base d'un dual encodeur (Lowe *et al.*, 2015). Pour cela, nous proposons une architecture simple, qui ne requiert pas de modules externes et entraînée de bout en bout. Notre système s'appuie sur la similarité entre le contexte de la conversation et la réponse candidate. À partir des vecteurs du contexte et de la réponse encodés par un dual encodeur LSTM (*Long Short-Term Mermory*) (Hochreiter & Schmidhuber, 1997), nous calculons leur produit vectoriel. Le vecteur résultant mesure la similarité entre le contexte et la réponse. Nous transformons ce vecteur de similarité en une probabilité avec une couche entièrement connectée et une fonction sigmoïde. Cette probabilité est utilisée par la suite pour ordonner les réponses candidates selon leur adéquation au contexte. Cette nouvelle méthode de calcul de similarité permet de capturer des propriétés sémantiques communes entre le contexte et la réponse. Nous avons évalué notre approche sur un large corpus de dialogues issus du canal *#Ubuntu* sur le Freenode IRC et nous avons suivi Lowe *et al.* (2015), Wu *et al.* (2016), Xu *et al.* (2017) et Wu *et al.* (2017) pour le choix des métriques d'évaluation : le Rappel@k et le Mean Recall Rank (MRR). Les résultats expérimentaux montrent des améliorations significatives [5].

La suite de cet article est organisée comme suit : nous résumons les travaux autour des systèmes conversationnels dans la section 2. Ensuite nous formalisons le problème et décrivons l'architecture de notre système dans la section 3. Les détails d'implémentation et les résultats expérimentaux sont donnés en section 4. Nous concluons dans la section 5 avec quelques perspectives.

5. Validées à l'aide d'un test de significativité.

2 État de l'art

Récemment, plusieurs travaux se sont orientés vers la construction de systèmes conversationnels à base de réseaux de neurones profonds. Dans ce cadre, la plupart des systèmes génératifs se basent sur l'architecture *séquence à séquence* de Sutskever *et al.* (2014) pour générer des dialogues (Vinyals & Le, 2015; Serban *et al.*, 2016; Sordoni *et al.*, 2015). Bien que ces systèmes génèrent des énoncés personnalisés pour chaque contexte de conversation, ils ont tendance à générer des réponses courtes et générales (Shao *et al.*, 2017; Li *et al.*, 2016). Ceci est dû essentiellement à la complexité de la tâche et au choix des fonctions objectifs qui entraîne un manque de diversité dans les réponses générées (Li *et al.*, 2016). En revanche, les systèmes d'ordonnancement de réponses sont capables de trouver des énoncés plus précis et syntaxiquement corrects dans le cas où ces énoncés figurent dans l'ensemble des réponses candidates. Ce type de système est au centre de nos intérêts dans le cadre de ce travail.

Lowe *et al.* (2015) ont proposé un système d'ordonnancement de réponses à base de *dual encodeur*. Le principe de ce système consiste à encoder le contexte et la réponse candidate séparément dans deux vecteurs. Le contexte consiste en la concaténation des tours de parole successifs dans l'historique de la conversation. Ensuite un score de similarité est calculé comme étant un produit de ces deux vecteurs et d'une matrice de paramètres appris par le système. Ce score est utilisé pour ordonner les réponses candidates. De plus, différentes variantes de cette approche à base de LSTM et de RNN (*Recurrent Neural Network*) ont été étudiées dans le même travail. Une extension de cette étude a été réalisée dans le travail de Kadlec *et al.* (2015) dans laquelle une approche ensembliste à base du dual encodeur de Lowe *et al.* (2015) a été déployée regroupant 11 LSTMs, 7 Bi-LSTMs et 10 CNNs. Une moyenne des scores de ces systèmes est calculée pour obtenir le score final de la réponse candidate.

Inspirés par le fonctionnement du cerveaux humain, Xu *et al.* (2017) ont incorporé dans leur travail des connaissances sur le domaine pour mieux modéliser le contexte et la réponse. Ils ont introduit pour la première fois une nouvelle cellule r-LSTM qui a une porte supplémentaire appelée *"Recall Gate"*. Cette cellule, comme son nom l'indique, sert à mémoriser les connaissances sur le domaine. D'abord ces connaissances sont obtenues grâce à une base de connaissance construite manuellement et qui permet d'obtenir des mots liés au domaine à partir du contexte et de la réponse candidate. En plus du contexte et de la réponse candidate, les informations liées au domaine sont encodées grâce à un encodeur r-LSTM en un vecteur qui résume toute la conversation. Ce vecteur est transformé en une probabilité utilisée comme score d'ordonnancement de réponses.

Wu *et al.* (2017) ont développé un système qui considère cette fois-ci les tours de parole séparément. Ils ont extrait deux types d'information de chaque tour de parole sous forme de deux matrices : la similarité au niveau des mots et des tours de parole. Grâce à une succession de convolution et de max-pooling, ces matrices de similarité ont été transformées en des vecteurs. Ensuite, ces vecteurs ont été accumulés grâce à un réseau de neurones récurrents à base d'unité GRU (Gated Recurrent Unit) (Chung *et al.*, 2014) afin d'obtenir un score de correspondance entre le contexte et la réponse.

Contrairement à tous ces travaux, Wang *et al.* (2013); Wu *et al.* (2016) se sont limités au dernier tour de parole. Wu *et al.* (2016) ont exploité le sujet de la conversation comme information supplémentaire afin d'améliorer la qualité de la réponse sélectionnée. Ils ont utilisé un modèle de sujets : le Twitter LDA (Zhao *et al.*, 2011) afin de générer un sujet pour le contexte et la réponse candidate. Le contexte, la réponse et leurs sujets respectifs ont été représentés par des plongements de mots et transformés en des vecteurs grâce à une convolution et au max-pooling. Ensuite ces vecteurs ont été appariés deux à deux grâce à des Réseaux de Tenseurs Neuronaux (*Neural Tensor Networks NTNs*) (Socher *et al.*,

2013) afin d'obtenir le score de la réponse. Bien que d'autres types d'information ont été pris en compte, cette restriction au dernier tour de parole est une hypothèse forte qu'il faudrait lever.

Dans ce travail, nous adoptons le premier système qui a abordé le problème d'ordonnancement de réponses avec une architecture neuronale : le dual encodeur de Lowe *et al.* (2015). Nous proposons une nouvelle approche de calcul du score de la réponse candidate. Contrairement à l'approche ensembliste de Kadlec *et al.* (2015) qui génère plusieurs paramètres à raffiner dans le cas où nous changeons de domaine d'application, notre approche est simple et facilement portable. Notre système ne requiert pas d'informations externes liées au domaine contrairement au système de Xu *et al.* (2017) ce qui favorise encore plus son adaptation à d'autres domaines. De plus le problème de reproductibilité des résultats de Wu *et al.* (2017), comme expliqué dans la section 4, ne nous a pas permis d'exploiter leur système et a motivé notre choix de nous appuyer sur l'architecture du système initial.

3 Modèle

Dans cette section, nous formalisons le problème auquel nous nous intéressons et nous décrivons l'architecture de notre système d'ordonnancement de réponses.

3.1 Formalisation du problème

Étant donné un contexte C de conversation entre deux utilisateurs sous forme d'une succession de n tours de paroles t_i tel que $C = \{t_1, t_2, t_3, ...t_n\}$. Le problème consiste à sélectionner le prochain tour de parole t_{n+1} appelé la réponse à ce contexte parmi un ensemble de m réponses possibles $t_{n+1} \in \{r_1, r_2, r_3, ..., r_m\}$. Nous définissons le problème comme étant un problème d'ordonnancement dans lequel nous classons les réponses candidates dans un ordre croissant de leurs pertinences vis-à-vis du contexte de la conversation. La réponse ayant le plus grand score est choisie comme étant le prochain tour de parole dans la conversation.

3.2 Architecture du modèle

Inspirés par le système de Lowe *et al.* (2015), nous proposons une architecture améliorée du dual encodeur à base de LSTM entraînée de bout en bout (Figure 2). Tout d'abord nous concaténons les tours de parole du contexte en gardant un simple marqueur de fin de tour. Nous ne mettons aucune restriction sur la taille du contexte en termes du nombre de tours de parole présents. L'idée de base consiste à représenter le contexte C et la réponse R d'abord en utilisant les plongements des mots. Ensuite ces plongements $e_1, e_2, .., e_j$ sont fournis dans l'ordre chronologique des mots à un encodeur. Cet encodeur consiste en un réseau de neurones récurrents à base de cellules LSTM, dont la couche cachée est mise à jour à chaque fois qu'un plongement de mots est donné en entrée. Ce processus est modélisé dans le cadre en Figure 2, il est similaire à celui déployé dans le système de Lowe *et al.* (2015). En sortie, nous récupérons la dernière couche cachée de l'encodeur C' et R' qui représente dans ce cas le contexte dans son ensemble et la réponse respectivement.

Lowe *et al.* (2015) calculent le score de la réponse candidate R par rapport au contexte C en multipliant C' par R' et par une matrice M de paramètres appris par le modèle. Dans notre approche, le score est calculé à partir d'un produit vectoriel P entre C' et R' qui reflète la similarité entre le

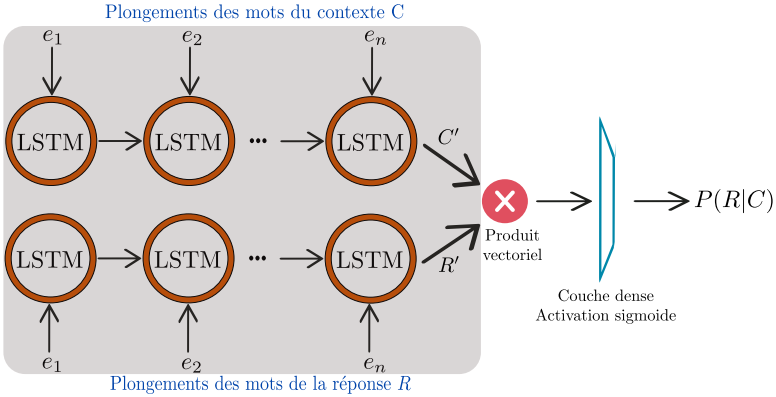


FIGURE 2: Architecture de notre système à base de dual encodeur

contexte et la réponse. Le résultat est transformé en une probabilité grâce à une fonction sigmoïde. Notre motivation réside dans le fait que dans une conversation le contexte et la réponse partagent des notions communes. Ces notions sont capturées d'abord par les plongements de mots et ensuite, grâce aux encodeurs et au calcul de la similarité, nous capturons des similarités sémantiques entre le contexte et la réponse.

Dans l'approche de base, Lowe *et al.* (2015) définissent la tâche de recherche du prochain tour de parole comme une tâche de génération. En plus des vecteurs C' et R', leur dual encodeur apprend une matrice M de paramètres qui sera utilisée pour générer implicitement une réponse R'' en multipliant C' par M (Équation 1).

$$R'' = C'^T \cdot M \tag{1}$$

Ensuite cette réponse R'' générée à partir du contexte de la conversation est comparée à la réponse que le système devrait prédire R'. La comparaison est effectuée grâce à un produit scalaire entre R'' et R' (Équation 2). Ce score de similarité est utilisé par la suite pour ordonner les réponses candidates.

$$Score = R'' \cdot R' \tag{2}$$

Dans l'approche que nous proposons dans ce travail, nous ordonnons les réponses candidates selon leur similarité sémantique directe avec le contexte de la conversation. Nous calculons cette similarité via un produit vectoriel entre le vecteur du contexte et le vecteur de la réponse que nous obtenons à partir de l'encodeur. Ce produit mesure l'intensité du lien entre chacune des réponses candidates et le contexte de la conversation. En conséquence, notre système apprend à ordonner les réponses en prenant en compte la sémantique partagée entre celles-ci et le contexte. Ceci explique l'amélioration des résultats que nous obtenons par rapport aux systèmes état de l'art de Lowe *et al.* (2015) et Kadlec *et al.* (2015).

4 Expériences et résultats

Dans cette section, nous présentons notre environnement expérimental ainsi que les résultats d'évaluation. Nous commençons par décrire le corpus sur lequel nous avons évalué notre système. Ensuite nous définissons les métriques d'évaluation que nous avons utilisées. Enfin nous discutons les résultats expérimentaux ainsi que les paramètres de notre système.

4.1 Corpus de dialogues d'Ubuntu

Dans le travail de Lowe *et al.* (2015) un large corpus de dialogues provenant de chat d'Ubuntu a été construit. Ce corpus comprend environ un million de conversations entre deux utilisateurs ayant au moins trois tours de parole. Ces conversations sont issues des logs du canal *#Ubuntu* sur le Freenode IRC (Internet Relay Chat) [6]. Les conversations sont du chat en anglais et traitent des sujets techniques divers. La première version (V1) de ce corpus comprenait quelques lacunes qui ont été corrigées plus tard dans la deuxième version (V2) [7]. La table 2 présente quelques statistiques et propriétés de la version 2 de ce corpus.

# énoncés (au total)	7 100 000
# tours de parole (au total)	5 139 574
# mots (au total)	100 000 000
Min. # tours de parole par dialogue	3
Moy. # tours de parole par dialogue	4,94
Moy. # mots par énoncé	10,34
# dialogues d'entraînement	1 000 000
# dialogues de test	18 920
# dialogues de validation	19 560

TABLE 2: Propriétés du corpus d'Ubuntu V2

Le corpus contient un million de dialogues pour l'entraînement, 19 560 pour la validation et 18 920 pour le test. Chaque élément d'entraînement est un triplet *(contexte, réponse, étiquette)*. L'étiquette est à "1" dans le cas où la réponse est le prochain tour de parole, "0" dans le cas contraire. Dans les ensembles de validation et de test, chaque élément est composé d'un contexte, d'une bonne réponse et de neuf mauvaises réponses (extraites aléatoirement d'autres conversations).

La tâche sur ce corpus consiste à ordonner la bonne réponse au rang supérieur par rapport aux neuf mauvaises réponses. Nous avons choisi d'évaluer notre système sur ce corpus pour essentiellement deux raisons. La première est liée à notre objectif de construire un système d'ordonnancement de réponses spécifique à un domaine qui est ici, l'assistance technique autour d'Ubuntu. La deuxième raison est le fait que plusieurs systèmes d'ordonnancement de réponses ont été évalués sur ce corpus, ce qui nous a aidé à préparer notre environnement d'évaluation et permis de comparer les résultats obtenus.

6. Sur la période 2004-2015 disponible sur `https://irclogs.ubuntu.com/`
7. Disponible sur `https://github.com/rkadlec/ubuntu-ranking-dataset-creator`

4.2 Métriques d'évaluation

L'évaluation des systèmes conversationnels est un domaine de recherche ouvert dans lequel il n'y a pas de métriques standards (Lowe *et al.*, 2017a; Liu *et al.*, 2016). Nous avons suivi Lowe *et al.* (2015), Wu *et al.* (2016), Xu *et al.* (2017) et Wu *et al.* (2017) dans l'utilisation du *Rappel@k*, du *Mean Recall Rank* (MRR) (Voorhees, 2001) comme métriques d'évaluation de notre système d'ordonnancement de réponses. Ces deux métriques mesurent la capacité du système à ordonner la bonne réponse avant les mauvaises réponses. Notons qu'en raison de la présence d'une seule bonne réponse pour chaque contexte dans notre corpus de dialogue d'Ubuntu, la *Mean Average Precision* (MAP) (Baeza-Yates & Ribeiro-Neto, 1999) et la *Précision@1* sont équivalentes aux MRR et Rappel@1 respectivement.

4.3 Systèmes état de l'art

Nous rapportons les résultats de quatre systèmes état de l'art auxquels nous comparons notre système. Nous avons vérifié la version du corpus d'Ubuntu sur laquelle chacun de ces systèmes a été évalué et nous rapportons les résultats dans le cas où il s'agit de la version V2. Dans le cas où le système a été évalué sur la V1 uniquement, nous l'avons ré-évalué en utilisant le code source des auteurs (si disponible) sur la V2. Les systèmes auxquels nous avons comparé notre système sont les suivants :

TF-IDF Nous rapportons les résultats de l'approche Term Frequency-Inverse Document Frequency présentée comme système état de l'art dans le travail de Lowe *et al.* (2015) et ré-évalué sur la V2 du corpus d'Ubuntu par Lowe *et al.* (2017b). Le contexte et chacune des réponses candidates sont représentés par un vecteur des scores TF-IDF des mots. Ensuite une similarité cosinus est calculée entre le vecteur du contexte et de la réponse afin d'obtenir un score de réponses.

RNN/LSTM dual encodeur Ces deux modèles ont été introduits dans le travail de Lowe *et al.* (2015) et ré-évalués sur la V2 du corpus par la suite dans le travail de Lowe *et al.* (2017b).

BiLSTM dual encodeur Ce système a été évalué par Kadlec *et al.* (2015) sur la V1 du corpus. Grâce au code source de Lowe *et al.* (2015), nous avons ré-évalué ce même système sur la V2 en utilisant les paramètres que les auteurs avaient décrits dans leur article.

4.4 Résultats

Dans la table 3, les trois premières lignes rapportent les résultats des systèmes de Lowe *et al.* (2017b). Le BiLSTM dans la ligne 4 est le système de Kadlec *et al.* (2015) que nous avons ré-évalué sur la V2. Notons bien que notre système surpasse les autres systèmes état de l'art et ceci sur toutes les métriques Rappel@k. Le Rappel@1 est une mesure forte de la capacité du système à ordonner la bonne réponse en premier parmi les 10 réponses candidates. De plus, nous remarquons que l'utilisation des cellules LSTM Bidirectionnelles (BiLSTM) dans notre système permet d'améliorer les résultats. Dans cette approche, nous obtenons deux vecteurs pour le contexte et la réponse grâce aux BiLSTM que nous concaténons pour obtenir un vecteur représentatif. Les résultats de la table 3 sont obtenus à partir d'une seule exécution des programmes pour laquelle l'entraînement converge sur l'ensemble de validation.

Comme expliqué dans la section 3, la différence dans la manière dont nous calculons la similarité entre le contexte et la réponse améliore significativement les résultats et nous permet de gagner environ 7, 6 et 2 points sur le Rappel@(1, 2 et 5) respectivement. Nous avons réalisé le test T de

Méthode	Rappel@1	Rappel@2	Rappel@5	MRR
TF-IDF (Lowe *et al.*, 2017b)	48,8 %	58,7 %	76,3 %	-
RNN Dual Encodeur (Lowe *et al.*, 2017b)	37,9 %	56,1 %	83,6 %	-
LSTM Dual Encodeur (Lowe *et al.*, 2017b)	55,2 %	72,1 %	92,4 %	-
BiLSTM Dual Encodeur (Kadlec *et al.*, 2015)	54,2 %	71,6 %	91,9 %	-
Similarité LSTM Dual Encodeur *(Sim LSTM DE)*	**62,9**† %	**78,5**† %	**95,2**† %	**76,1**† %
Similarité BiLSTM Dual Encodeur *(Sim BiLSTM DE)*	**63,7**† %	**79,1**† %	**95,2**† %	**76,7**† %

TABLE 3: Résultats de l'évaluation en utilisant les métriques Rappel@k et MRR.

Student (Student, 1908) pour évaluer la significativité des résultats. Les résultats de ce test montrent une nette significativité des écarts entre notre approche et celle de base que nous avons améliorée.

4.5 Évaluation de l'impact des plongement de mots

Dans ce travail, nous nous sommes intéressés à l'étude de l'impact des plongements de mots sur les performances de notre système. Pour cela, nous avons réalisé un ensemble d'expérimentations que nous résumons dans la table 4. Nous avons comparé 3 variantes de notre système. Pour chacune des variantes, nous avons changé les poids initiaux de la couche des plongements des mots. Nous avons fixé la taille de ces vecteurs à 300 et nous avons exploré l'impact de l'affinement des poids de la couche des plongements de mots comparé au non affinement (gèle) des poids durant la phase d'entraînement.

Les modèles de plongements de mots utilisés dans cette étude pour initialiser les poids de la couche des plongements de mots sont les suivants :

— **Word2Vec** : Nous avons entraîné word2vec (Mikolov *et al.*, 2013) sur l'ensemble d'entraînement en utilisant Gensim (Řehůřek & Sojka, 2010). Taille du vocabulaire = 770k.

— **FastText** : Nous avons utilisé des vecteurs de plongement de mots pré-entraînés sur Wikipedia en utilisant FastText [8] (Bojanowski *et al.*, 2017). Taille du vocabulaire = 2.5M.

— **Glove** : Nous avons utilisé des vecteurs pré-entraînés avec Glove (Pennington *et al.*, 2014) [9] sur Common Crawl Corpus [10]. Taille du vocabulaire = 2.2M.

Système	Rappel@1	Rappel@2	Rappel@5	MRR
Word2Vec-gelé	62,2 %	77,8 %	94,6 %	75,5 %
Word2Vec-affiné	63,3 %	78,4 %	94,9 %	76,2 %
FastText-gelé	58,9 %	75,1 %	94,2 %	73,2 %
FastText-affiné	61,7 %	77,8 %	94,7 %	75,2 %
Glove-gelé	62,5 %	78,0 %	94,8 %	75,7 %
Glove-affiné	62,9 %	78,5 %	95,2 %	76,1 %

TABLE 4: Évaluation de l'impact des plongements de mots sur les performances de notre système.

†. Scores significativement différents du dual encodeur de base au seuil de confiance de 0,01 selon le test T de Student.

8. https://s3-us-west-1.amazonaws.com/fasttext-vectors/wiki.en.vec

9. http://nlp.stanford.edu/data/glove.840B.300d.zip

10. http://commoncrawl.org/the-data/

En se basant sur les résultats, nous observons que pour tous les systèmes, l'affinement des poids de la couche des plongements de mots permet d'améliorer les résultats. De plus, ces plongements pré-entraînés avec Glove donnent les meilleurs résultats. Nous avons remarqué que l'entraînement des plongements de mots au niveau des mots dans le cas de Glove et Word2Vec donnent de meilleurs résultats en comparaison à ceux obtenus au niveau des caractères dans le cas de FastText.

4.6 Évaluation qualitative et quantitative des résultats

En plus de l'évaluation de notre système via les métriques spécifiques aux systèmes d'ordonnancement de réponse, nous avons mené une étude qualitative et quantitative pour analyser les résultats obtenus. Dans ce but, nous avons entraîné le système de Lowe *et al.* (2015) à l'aide de leur code source jusqu'à obtenir des scores similaires à ceux rapportés dans leur publication. Nous avons ensuite comparé les prédictions obtenues par leur système et le nôtre.

Contexte	- Hello .. Is it possible to disable GPG check for a specific APT repository ? - Why would you ever need to do that - It's for a custom repository in enterprise environment. But that's unimportant isn't it. Was that a statement that it's not possible ?		
LSTM DE	Sim LSTM DE	Étiquette	Réponse
0,06	**0,87**	1	3rd party repo ? PPA ? what is it ?
0,29	0,25	0	Find it sticky edge
0,17	0,40	0	That response doesn't help me in the slightest

Contexte	- How can I remount a drive as read/write ? - mount -o rw /dev/whatever /whever I belive theres a remount option i think - Thanks - I'd say check the mount man page also. I forgot the syntax for the remount option		
LSTM DE	Sim LSTM DE	Étiquette	Réponse
0,96	0,49	1	Okay
0,62	**0,88**	0	Thats sound like a good idea find out I'm missing authz_hos somehow
0,14	0,87	0	Thanks I will read that

Contexte	- Is there a length limitation on the hostname in SSH ? - 255 char for the FQDN - FQDN ?		
LSTM DE	Sim LSTM DE	Étiquette	Réponse
0,99	**0,94**	1	Full Qualify Domain Name : mycomputer.kitchen.myhouse.com
0,01	0,27	0	Alright good luck
0,01	0,08	0	You have to do it once the bios hand off to grub

Contexte	- Is there a script that can generate a live cd iso of your currently run ubuntu hdd install ? - Remastersys - Have you use it ?		
LSTM DE	Sim LSTM DE	Étiquette	Réponse
0,05	0,18	1	I hadn't have much luck with it, but that is a while ago (2 years)
0,88	**0,71**	0	It can
0,91	0,56	0	Not for me I doubt I could figure it out to be honest, but any theme installations come to mind as a guess

TABLE 5: Exemples d'accord et de désaccord de prédictions entre notre système (Sim LSTM DE) et le système de base de Lowe *et al.* (2015) (LSTM DE). Les scores en gras sont les scores les plus élevés attribués par le système.

La table 5 représente quelques exemples extraits à partir de l'ensemble de test. Chaque exemple est composé d'un contexte qui contient entre trois et quatre tours de parole, trois réponses candidates

avec leurs étiquettes (1 : bonne réponse, 0 : mauvaise réponse) et le score de prédiction obtenu par chacun des systèmes. De plus, les statistiques en nombre de cas d'accord et de désaccord entre les deux systèmes sont données dans la table 6. Dans le premier exemple, notre système attribue le score le plus élevé à la bonne réponse contrairement au dual encodeur de base. Malgré la difficulté du choix de la réponse car aucune des réponses ne partage explicitement des mots avec le contexte, notre système a pu capturer des relations sémantiques entre *repo* et *repository*, *PPT* et *APT*.

Sim LSTM DE \ LSTM DE	Réussite	Échec
Réussite	7437	4476
Échec	2143	4864

TABLE 6: Statistiques sur le nombre de cas d'accord et de désaccord entre les deux systèmes testés. Rappelons que la taille de l'ensemble de test est de 18 920 dialogues.

Le deuxième exemple représente le cas où le dual encodeur de Lowe *et al.* (2015) a pu retrouver la bonne réponse contrairement à notre système. Notons bien que même si notre système n'a pas réussi à retrouver la bonne réponse en premier, il a attribué le deuxième meilleur score à la troisième réponse. Cette réponse peut très bien remplacer la bonne réponse sans altérer le sens. Dans le troisième exemple, les deux systèmes retrouvent la bonne réponse avec des scores très élevés. Le dernier exemple représente un cas de figure dans lequel les deux systèmes ont échoué. Toutefois nous pensons que les deux réponses marquées avec une étiquette *"0"* sont des réponses potentielles au contexte.

Nous voyons donc dans la table 6 que notre système obtient des meilleurs résultats que LSTM DE (ce qui corrobore les résultats donnés dans la table 3). Mais il reste quand même 2143 cas où LSTM DE a trouvé la bonne réponse alors que le nôtre en a donné une mauvaise. Réaliser l'analyse de ces cas pourrait permettre de comprendre les lacunes de notre système afin de l'améliorer. Dans le but d'obtenir une analyse plus précise, il serait aussi intéressant de prendre en compte parmi les mauvaises réponses données par les systèmes, celles qui pourraient être considérées complétement cohérentes par rapport au contexte.

4.7 Paramètres du système

Les plongements de mots ont été initialisés avec Glove (Pennington *et al.*, 2014) préalablement entraînés sur Common Crawl Corpus puis affinés durant l'entraînement [11]. Les seuls prétraitements effectués sont la tokenisation, la racinisation et ensuite la lemmatisation (dans l'ordre) disponibles au moment du téléchargement du corpus à l'aide de NLTK (Loper & Bird, 2002). Les paramètres du système ont été mis à jour avec un gradient de descente stochastique avec l'algorithme Adam (Kingma & Ba, 2015). Tout le modèle a été entraîné sur un seul GPU Titan X.

Le taux d'apprentissage initial est de 0,001 et les paramètres d'Adam β_1 et β_2 sont 0,9 et 0,999 respectivement. Comme stratégie de régularisation, nous avons utilisé le *"early-stopping"* et pour entraîner nos modèles, nous avons utilisé un mini-batch de taille 256. La taille des plongements de mots de la couche cachée du LSTM est de 300. La taille du contexte et de la réponse a été réduite à 160 mots. Nous avons implémenté notre système avec Keras (Chollet *et al.*, 2015) ayant Tensorflow (Abadi *et al.*, 2015) comme backend [12].

11. Notons que nous avons entraîné les plongements de mots sur l'ensemble d'entraînement sans amélioration des résultats.
12. Nous mettons en ligne le code source qui permet de reproduire nos résultats sur `https://github.com/basma-b/`

5 Conclusion et perspectives

Nous avons proposé dans ce travail une approche d'ordonnancement de réponses dans les conversations écrites à base de dual encodeur. Les résultats expérimentaux montrent que notre approche apporte des améliorations significatives en comparaison aux approches de l'état de l'art. La nouvelle méthode basée sur la similarité sémantique entre le contexte et la réponse pour le calcul du score de pertinence de chaque réponse permet en particulier de mieux associer le contexte à la bonne réponse.

Par la suite, nous souhaitons d'abord ré-évaluer les autres approches état de l'art (Wu *et al.*, 2016, 2017; Xu *et al.*, 2017) sur le même jeu de données. Nous avons aussi comme objectif d'améliorer la représentation du contexte de la conversation en considérant, cette fois-ci, les tours de paroles de manière distincte au lieu de les concaténer simplement. Nous analyserons en détail les statistiques obtenus pour mieux comprendre les raisons de réussite et d'échec de notre système et proposer d'éventuelles améliorations. De plus, nous introduirons le mécanisme d'attention pour apprendre une meilleure représentation du contexte. Nous souhaitons aussi évaluer l'impact des prétraitements tels que l'élimination des mots outils et le filtrage des urls, numéros, etc. sur cette approche. En outre, une évaluation de nos méthodes sur de plus grands corpus de différentes langues tels que Baidu Tieba (Wu *et al.*, 2016) et Douban (Wu *et al.*, 2017) est prévue.

6 Remerciements

Ce travail a été partiellement financé par le projet ANR 2016 PASTEL CE33-0007 [13]. Nous remercions les relecteurs anonymes pour leurs nombreuses remarques qui ont été utiles pour l'amélioration du contenu de l'article.

Références

ABADI M., AGARWAL A., BARHAM P., BREVDO E., CHEN Z., CITRO C., CORRADO G. S., DAVIS A., DEAN J., DEVIN M., GHEMAWAT S., GOODFELLOW I., HARP A., IRVING G., ISARD M., JIA Y., JOZEFOWICZ R., KAISER L., KUDLUR M., LEVENBERG J., MANÉ D., MONGA R., MOORE S., MURRAY D., OLAH C., SCHUSTER M., SHLENS J., STEINER B., SUTSKEVER I., TALWAR K., TUCKER P., VANHOUCKE V., VASUDEVAN V., VIÉGAS F., VINYALS O., WARDEN P., WATTENBERG M., WICKE M., YU Y. & ZHENG X. (2015). TensorFlow : Large-scale machine learning on heterogeneous systems. Software available from tensorflow.org.

BAEZA-YATES R. A. & RIBEIRO-NETO B. (1999). *Modern Information Retrieval*. Boston, MA, USA : Addison-Wesley Longman Publishing Co., Inc.

BOJANOWSKI P., GRAVE E., JOULIN A. & MIKOLOV T. (2017). Enriching word vectors with subword information. *Transactions of the Association of Computational Linguistics (TACL)*, **5**, 135–146.

CHOLLET F. *et al.* (2015). Keras. `https://github.com/keras-team/keras`.

dual_encoder_udc

13. http://www.agence-nationale-recherche.fr/?Projet=ANR-16-CE33-0007

CHUNG J., GULCEHRE C., CHO K. & BENGIO Y. (2014). Empirical evaluation of gated recurrent neural networks on sequence modeling. In *Workshop on Deep Learning and Representation Learning at the 28th Annual conference on Advances in Neural Information Processing Systems (NIPS'14)*, Montreal, Canada.

HOCHREITER S. & SCHMIDHUBER J. (1997). Long short-term memory. *Neural computation*, **9**(8), 1735–1780.

KADLEC R., SCHMID M. & KLEINDIENST J. (2015). Improved deep learning baselines for ubuntu corpus dialogs. In *Workshop on Machine Learning for Spoken Language Understanding and Interaction at the 29th Annual Conference on Neural Information Processing Systems (NIPS'15)*, Montreal, Canada.

KINGMA D. & BA J. (2015). Adam : A method for stochastic optimization. In *Proceedings of the 3rd International Conference for Learning Representations (ICLR'15)*, San Diego, CA, USA.

LI J., GALLEY M., BROCKETT C., GAO J. & DOLAN B. (2016). A diversity-promoting objective function for neural conversation models. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics : Human Language Technologies (NAACL'16)*, p. 110–119, San Diego, CA, USA.

LI X., CHEN Y.-N., LI L., GAO J. & CELIKYILMAZ A. (2017). End-to-end task-completion neural dialogue systems. In *Proceedings of the Eighth International Joint Conference on Natural Language Processing (AFNLP'17)*, p. 733–743, Taipei, Taiwan.

LIU C.-W., LOWE R., SERBAN I., NOSEWORTHY M., CHARLIN L. & PINEAU J. (2016). How not to evaluate your dialogue system : An empirical study of unsupervised evaluation metrics for dialogue response generation. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing (EMNLP'16)*, p. 2122–2132, Austin, Texas.

LOPER E. & BIRD S. (2002). Nltk : The natural language toolkit. In *Proceedings of the ACL Workshop on Effective Tools and Methodologies for Teaching Natural Language Processing and Computational Linguistics (ETMTNLP'02)*, p. 63–70, Stroudsburg, PA, USA.

LOWE R., NOSEWORTHY M., SERBAN I. V., ANGELARD-GONTIER N., BENGIO Y. & PINEAU J. (2017a). Towards an automatic turing test : Learning to evaluate dialogue responses. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (ACL'17)*, p. 1116–1126, Vancouver, Canada.

LOWE R., POW N., SERBAN I. & PINEAU J. (2015). The ubuntu dialogue corpus : A large dataset for research in unstructured multi-turn dialogue systems. In *Proceedings of the 16th Annual Meeting of the Special Interest Group on Discourse and Dialogue (SIGDIAL'15)*, p. 285–294, Prague, Czech Republic.

LOWE R. T., POW N., SERBAN I. V., CHARLIN L., LIU C.-W. & PINEAU J. (2017b). Training end-to-end dialogue systems with the ubuntu dialogue corpus. *Dialogue & Discourse*, **8**(1), 31–65.

MIKOLOV T., CHEN K., CORRADO G. & DEAN J. (2013). Efficient estimation of word representations in vector space. In *Proceedings of the International Conference on Learning Representations (ICLR'13)*, p. 1–12, Scottsdale, Arizona.

PENNINGTON J., SOCHER R. & MANNING C. (2014). Glove : Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP'14)*, p. 1532–1543, Doha, Qatar.

ŘEHŮŘEK R. & SOJKA P. (2010). Software Framework for Topic Modelling with Large Corpora. In *Worshop on New Challenges for NLP Frameworks at the 7th edition of the Language Resources and Evaluation Conference (LREC'10)* , p. 45–50, Valletta, Malta.

SERBAN I. V., SORDONI A., BENGIO Y., COURVILLE A. & PINEAU J. (2016). Building end-to-end dialogue systems using generative hierarchical neural network models. In *Proceedings of the 30th AAAI Conference on Artificial Intelligence (AAAI'16)*, p. 3776–3783, Phoenix, AZ, USA.

SHAO Y., GOUWS S., BRITZ D., GOLDIE A., STROPE B. & KURZWEIL R. (2017). Generating high-quality and informative conversation responses with sequence-to-sequence models. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing (EMNLP'17)*, p. 2210–2219, Copenhagen, Denmark.

SOCHER R., CHEN D., MANNING C. D. & NG A. (2013). Reasoning with neural tensor networks for knowledge base completion. In *Proceedings of the 26th international conference on Advances in Neural Information Processing Systems (NIPS'13)*, p. 926–934. Lake Tahoe, NV, USA.

SORDONI A., BENGIO Y., VAHABI H., LIOMA C., GRUE SIMONSEN J. & NIE J.-Y. (2015). A hierarchical recurrent encoder-decoder for generative context-aware query suggestion. In *Proceedings of the 24th ACM International Conference on Information and Knowledge Management (CIKM'15)*, p. 553–562, Melbourne, Australia.

STUDENT (1908). The probable error of a mean. *Biometrika*, p. 1–25.

SUTSKEVER I., VINYALS O. & LE Q. V. (2014). Sequence to sequence learning with neural networks. In *Proceedings of the 2014 conference on Advances in Neural Information Processing Systems (NIPS'14)*, p. 3104–3112. Montreal, Canada.

VINYALS O. & LE Q. (2015). A neural conversational model. In *Workshop on Deep Learning at the 31 st International Conference on Machine Learning (ICML'15)*, Lille, France.

VOORHEES E. M. (2001). The trec question answering track. *Natural Language Engineering*, **7**(4), 361–378.

WANG H., LU Z., LI H. & CHEN E. (2013). A dataset for research on short-text conversations. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP'13)*, p. 935–945, Seattle, WA, USA.

WEN T.-H., VANDYKE D., MRKŠIĆ N., GASIC M., ROJAS BARAHONA L. M., SU P.-H., ULTES S. & YOUNG S. (2017). A network-based end-to-end trainable task-oriented dialogue system. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics (EACL'17)*, p. 438–449, Valencia, Spain.

WU Y., WU W., LI Z. & ZHOU M. (2016). Response selection with topic clues for retrieval-based chatbots. *arXiv preprint arXiv :1605.00090*.

WU Y., WU W., XING C., ZHOU M. & LI Z. (2017). Sequential matching network : A new architecture for multi-turn response selection in retrieval-based chatbots. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (ACL'17)*, p. 496–505, Vancouver, Canada.

XU Z., LIU B., WANG B., SUN C. & WANG X. (2017). Incorporating loose-structured knowledge into conversation modeling via recall-gate lstm. In *Proceedings of the International Joint Conference on Neural Networks (IJCNN'17)*, p. 3506–3513, Anchorage, AK, USA.

ZHAO W. X., JIANG J., WENG J., HE J., LIM E.-P., YAN H. & LI X. (2011). Comparing twitter and traditional media using topic models. In *Proceedings of the 33rd European Conference on Advances in Information Retrieval (ECIR'11)*, p. 338–349, Berlin, Heidelberg.

Intégration de contexte global par amorçage pour la détection d'événements

Dorian Kodelja Romaric Besançon Olivier Ferret
CEA, LIST, Laboratoire Vision et Ingénierie des Contenus, Gif-sur-Yvette, F91191 France
dorian.kodelja, romaric.besancon, olivier.ferret@cea.fr

RÉSUMÉ

Les approches neuronales obtiennent depuis plusieurs années des résultats intéressants en extraction d'événements. Cependant, les approches développées dans ce cadre se limitent généralement à un contexte phrastique. Or, si certains types d'événements sont aisément identifiables à ce niveau, l'exploitation d'indices présents dans d'autres phrases est parfois nécessaire pour permettre de désambiguïser des événements. Dans cet article, nous proposons ainsi l'intégration d'une représentation d'un contexte plus large pour améliorer l'apprentissage d'un réseau convolutif. Cette représentation est obtenue par amorçage en exploitant les résultats d'un premier modèle convolutif opérant au niveau phrastique. Dans le cadre d'une évaluation réalisée sur les données de la campagne TAC 2017, nous montrons que ce modèle global obtient un gain significatif par rapport au modèle local, ces deux modèles étant eux-mêmes compétitifs par rapport aux résultats de TAC 2017. Nous étudions également en détail le gain de performance de notre nouveau modèle au travers de plusieurs expériences complémentaires.

ABSTRACT

Integrating global context via bootstrapping for event detection.

Over the last few years, neural models developed for event extraction have reached an interesting level of results. However, their application is generally limited to sentences, which is sufficient for identifying certain types of events but too limited in terms of scope for disambiguating some occurrences of events. In this article, we propose to integrate in a convolutional neural network the representation of contexts beyond the level of sentences. This representation is built following a bootstrapping approach by exploiting an intra-sentential convolutional model. Within the evaluation framework of TAC 2017, we show that our global model significantlty outperforms the intra-sentential model while the two models are competitive with the results obtained by TAC 2017 participants. We furthermore analyze the gain of our proposed model through supplementary experiments.

MOTS-CLÉS : Détection d'événement, réseau de neurones convolutifs, contexte discursif.

KEYWORDS: Event detection, convolutional neural networks, discourse context.

1 Introduction

L'extraction d'événements supervisée consiste à identifier au sein d'un document les occurrences de types d'événements préalablement définis. Ces événements sont caractérisés par des interactions entre plusieurs entités y tenant des rôles spécifiques. Cette tâche est le plus souvent décomposée en plusieurs tâches de classification successives pour identifier d'abord la mention d'un événement puis ses arguments. Nous nous intéressons ici à la première étape, aussi appelée détection d'événements ou détection de mentions événementielles, tâche qui consiste à identifier dans le texte le ou les mots indiquant le plus clairement la présence d'un événement.

Les approches actuelles sont majoritairement fondées sur des modèles neuronaux, qu'il s'agisse de modèles convolutifs (Chen *et al.*, 2015; Nguyen & Grishman, 2015), récurrents (Nguyen *et al.*, 2016a) ou combinant les deux approches (Feng *et al.*, 2016). Le problème est alors modélisé sous forme d'une tâche de classification pour chaque mot du document. Les meilleurs systèmes des campagnes d'évaluation récentes sur le sujet relèvent par ailleurs tous de ce même paradigme.

Le jeu de données que nous utilisons par la suite est annoté selon la taxonomie Rich ERE (Song *et al.*, 2015). Ce schéma d'annotation distingue 38 sous-types d'événements répartis en 9 types. Ces types d'événements couvrent un large champ d'interactions possibles, allant des transactions financières aux conflits armés en passant par les correspondances écrites ou les licenciements. L'étendue et la finesse de cette modélisation amènent à distinguer plusieurs sources d'erreurs de détection des événements. L'une d'elles réside dans l'ambiguïté des marqueurs. Par exemple, la polysémie d'un verbe peut être source de confusion entre plusieurs types d'événements. Le mot "*fired*" peut ainsi faire référence à un coup de feu, indicateur d'un événement de type "*Conflict-Attack*", ou signifier "licencier" et indiquer un événement de type "*Personnel-End-Position*". Sur un autre plan, la proximité des sous-types d'événements appartenant à un même type peut rendre ceux-ci distinguables seulement par une compréhension fine de leur contexte et constitue de ce fait une autre source d'erreurs possible. Ainsi, le type d'événement "*Contact*" distingue les sous-types "*Contact-Broadcast*" lorsque la communication est à sens unique, "*Contact-Meet*" pour une rencontre physique entre plusieurs personnes, "*Contact-Correspondance*" s'il s'agit d'un échange à distance et "*Contact-Contact*" quand aucun sous-type plus spécifique ne correspond.

On voit ici l'importance de la prise en compte du contexte pour résoudre les différentes ambiguïtés possibles entre différents types. Cependant, si les systèmes présentés précédemment parviennent à identifier correctement une grande partie des mentions d'événements, des ambiguïtés subsistent lorsque le contexte local n'est pas assez informatif pour permettre au modèle de discriminer convenablement un type d'événements d'un autre ou de l'absence d'événement. On peut ainsi distinguer deux types d'ambiguïtés en fonction de la portée du contexte nécessaire à leur résolution :

— « Le rappeur lyonnais **déclara** "Les instruments c'était mieux à vent" aux micros de France Inter. »
— « Les **départs** se multiplient chez l'opérateur téléphonique. [...] Plus de 200 démissions ont ainsi été reçues le mois dernier. »

Dans le premier exemple, l'ambiguïté fréquente entre les types d'événements *Contact-contact* et *Contact-broadcast* peut être résolue en identifiant que la communication s'adresse à un média et se fait donc à sens unique. Mais la distance entre *déclara* et *France Inter* peut rendre cette information difficile à exploiter bien qu'elle soit locale à la phrase. Les modèles, en particulier convolutifs, n'exploitent en pratique qu'un contexte très restreint autour de la mention candidate à étiqueter

pour prendre une décision, même lorsque le contexte fourni est plus large. C'est ici un problème de désambiguïsation intra-phrastique. Dans le second exemple, le contexte local de la phrase n'indique pas clairement que *départ* fait référence à un événement de type *End_Position*. Mais la thématique des licenciements est clairement identifiable plus loin dans le document. Il s'agit ici d'un problème de désambiguïsation inter-phrastique.

Pour réaliser ces désambiguïsations, il est nécessaire d'exploiter un contexte plus global. Duan *et al.* (2017) mettent en avant l'intérêt d'une telle exploitation afin de prendre en compte la cohérence interne des documents sur le plan thématique : un document traitant d'un conflit armé présentera plus d'événements de type *Die* ou *Attack* que de naissances. Ils proposent de fournir en entrée d'un BiLSTM une représentation distribuée du document apprise de manière non supervisée (Le & Mikolov, 2014). Ce contexte global n'est donc pas spécialisé pour la tâche cible et n'est en outre adapté qu'au problème de désambiguïsation inter-phrastique. Notre approche vise au contraire à apprendre une représentation des documents en lien avec la tâche cible et ce, pour les deux cas de figure identifiés précédemment. Nous utilisons pour ce faire une méthode d'amorçage, en définissant un premier modèle à un niveau local (niveau des mots) et en l'appliquant à l'ensemble du document. Les prédictions locales ainsi réalisées sont agrégées pour obtenir un vecteur de contexte pour chaque document. Ces vecteurs sont alors intégrés à un nouveau modèle exploitant ce contexte.

Dans la suite de cet article, nous présentons dans un premier temps nos modèles local et global à la section 2. Puis, dans la section 3.2, nous étudions l'influence de la taille du contexte local sur les performances et observons l'incapacité du modèle de base à exploiter l'information distante, démontrant ainsi l'intérêt de l'intégration d'un contexte distant. La section 3.3 compare les différents types de représentations du contexte global utilisables. Enfin, nous évaluons à la section 3.4 différentes modalités d'intégration de cette représentation globale au modèle local et les confrontons à plusieurs baselines sur les données de la campagne d'évaluation TAC Event Nugget 2017. Ces expériences montrent que ce nouveau modèle obtient des gains significatifs par rapport au modèle local *baseline* et s'avère compétitif par rapport à d'autres approches neuronales. En dernier lieu, nous étudions plus en détail l'apport de la représentation globale à la section 4.

2 Description de l'approche

Comme nous l'avons vu en introduction, la détection d'événements consiste à identifier dans un texte les mentions d'événements et leur associer un type selon une taxonomie préalablement établie. Dans cet article, nous nous appuyons sur les 38 types d'événements de la taxonomie DEFT Rich ERE utilisée dans le cadre des campagnes TAC. Dans les annotations liées à cette taxonomie, les mentions d'événements étant en grande majorité des mots simples (Reimers & Gurevych, 2017), nous choisissons d'aborder le problème non pas comme une tâche d'annotation de séquences mais comme une tâche de classification multi-classe de mots. Ce choix est d'un impact négatif négligeable mais simplifie la modélisation et permet l'introduction d'un vecteur de positions contribuant grandement aux performances (Nguyen *et al.*, 2016b). Enfin, dans la continuité des approches neuronales récentes, nous nous plaçons à l'échelle intra-phrastique. La tâche est alors envisagée comme un problème de classification multi-classe.

Nous présentons à la figure 1 la procédure générale d'intégration du contexte à un modèle convolutif de détection d'événements. Un premier modèle $\text{CNN}_{\text{local}}$ est entraîné pour associer des étiquettes d'événement à chaque mot d'un document. Ces étiquettes sont agrégées au niveau d'un contexte

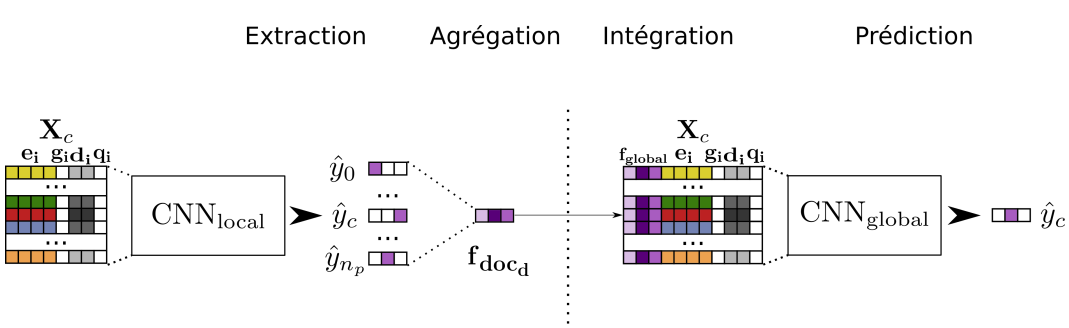


FIGURE 1 – Principe d'intégration du contexte par amorçage. $\hat{y}_c$ est la prédiction du modèle pour la mention courante, $\hat{y}_0$, la prédiction pour la première mention trouvée du document et $\hat{y}_{n_p}$ la dernière mention trouvée du document.

(dans la figure, ce contexte est le document) et ajoutées en entrée d'un nouveau modèle $\mathrm{CNN}_{\mathrm{global}}$. Nous présentons plus en détail ces modèles local et global dans les sections suivantes.

2.1 Modèle local de détection d'événements

Notre modèle de détection d'événements au niveau local s'appuie sur un réseau de neurones convolutif inspiré de l'architecture proposée par (Nguyen *et al.*, 2016b). Nous considérons successivement chaque mot de chaque phrase en tant que mention candidate. Cette mention est représentée par un contexte local de taille fixe centré sur ce mot. Si le contexte local dépasse les limites de la phrase courante, un token spécial est utilisé pour compléter la séquence. Soit i_c l'index de la mention candidate et w la taille de la fenêtre. On définit $\mathbf{t}_c = [i_{c-w}, i_{c-w+1}, \ldots, i_c, \ldots, i_{c+w-1}, i_{c+w}]$ le vecteur des index du contexte local centré sur i_c . Ce vecteur d'index est transformé en une matrice de réels $\mathbf{X}_c = [\mathbf{x_{c-w}}, \mathbf{x_{c-w+1}}, \ldots, \mathbf{x_c}, \ldots, \mathbf{x_{c+w-1}}, \mathbf{x_{c+w}}]$ en remplaçant chaque index i par une représentation $\mathbf{x_i} = [\mathbf{e_i}, \mathbf{d_i}, \mathbf{g_i}, \mathbf{q_i}]$ obtenue en combinant les différentes représentations suivantes :

Plongement de mot $\mathbf{e_i}$ Cette représentation distribuée du mot t_i est pré-entraînée sur un large corpus pour capter des informations sémantiques et syntaxiques à propos de ce mot (Mikolov *et al.*, 2013).

Vecteur de position $\mathbf{d_i}$ Ce vecteur encode la position relative i du mot t_i par rapport au candidat t_0.

Vecteur des dépendances syntaxiques $\mathbf{g_i}$ Ce vecteur a une dimension correspondant au nombre de dépendances considérées. Si une dépendance d'un certain type existe entre t_i et t_0, la dimension correspondante du vecteur est égale à 1. Dans nos expériences, nous utilisons les dépendances de base (*basic dependencies*) fournies par l'outil Stanford CoreNLP (Manning *et al.*, 2014).

Vecteur de syntagme $\mathbf{q_i}$ Ce vecteur encode le type de constituant syntaxique dont le token fait partie sous la forme d'une annotation IOB fournie par un *chunker* [1]. Cette représentation est construite à partir de l'arbre syntaxique fourni par l'outil Stanford CoreNLP.

À partir de cette matrice d'entrée $\mathbf{X}_c$, nous appliquons une couche de convolution constituée de plusieurs filtres de tailles différentes. Une couche de *global max-pooling* est ensuite appliquée afin d'obtenir une seule valeur pour chaque filtre. Nous obtenons ainsi une représentation du candidat dans son contexte local apprise par le réseau convolutif. Cette représentation locale $\mathbf{f_{softmax}} = [\mathbf{f_{pooling}}]$ est ensuite fournie en entrée d'une couche de neurones entièrement connectée (*fully connected*) dotée d'un softmax. Ce dernier étage permet de calculer la distribution de probabilités des différentes

1. https://github.com/mgormley/concrete-chunklink

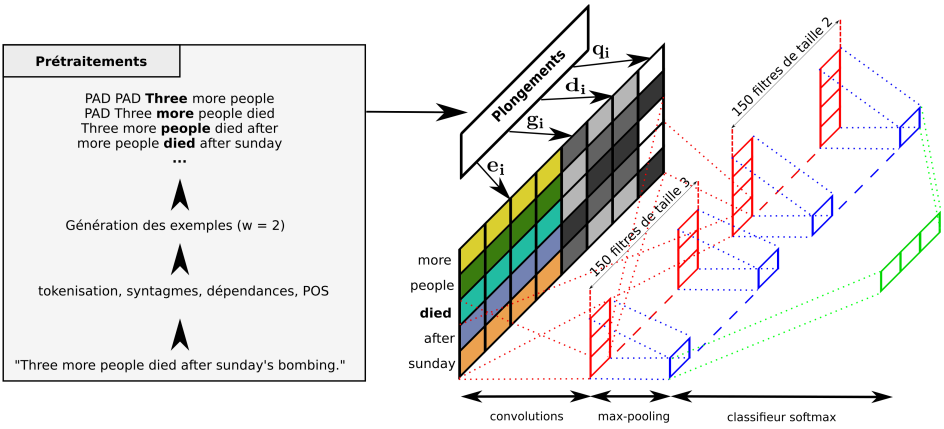


FIGURE 2 – Architecture du modèle $\text{CNN}_{\text{local}}$

classes d'événements pour le candidat et d'en déduire un label unique $\hat{y}_c$ en prenant la classe de probabilité maximale. Pour améliorer la généralisation, un dropout est appliqué sur la couche d'entrée du réseau. La figure 2 représente de manière détaillée l'architecture du $\text{CNN}_{\text{local}}$.

2.2 Intégration de contexte dans un modèle global

Afin d'augmenter les performances de notre modèle convolutif, nous proposons d'intégrer une information de contexte plus large, sous la forme d'une représentation globale focalisée sur la tâche d'extraction d'événement en utilisant un principe d'amorçage. Pour ce faire, nous réalisons un premier entraînement du modèle local présenté précédemment. Nous utilisons ensuite ce modèle pour extraire $\hat{y}_c$ pour chaque mot du corpus. Nous agrégeons alors $\hat{y}_c$ par *sum-pooling*, ce qui équivaut à construire un histogramme des différents types d'événements détectés. Nous réalisons cette agrégation à trois niveaux différents : à l'échelle de chaque phrase (*phrase*), d'un contexte de trois phrases centré sur la phrase courante (*large*) ou à l'échelle du document (*doc*).

Nous utilisons la notation suivante pour désigner les différentes configurations possibles de contexte global d'une phrase : $\mathbf{f_{global}} = \mathbf{f_{[doc/large/phrase]}}$. Le contexte global $\mathbf{f_{global}}$ peut être intégré au niveau de la matrice d'entrée $\mathbf{X}_c$ en redéfinissant $\mathbf{x_i} = [\mathbf{e_i}, \mathbf{d_i}, \mathbf{g_i}, \mathbf{q_i}, \mathbf{f_{global}}]$ ou concaténé avant la couche entièrement connectée : $\mathbf{f_{softmax}} = [\mathbf{f_{pooling}}, \mathbf{f_{global}}]$. On distinguera ainsi six modèles en fonction des niveaux d'agrégation et d'intégration avec la notation $\text{CNN}_{\text{[doc/large/phrase]-[plongement/softmax]}}$.

3 Expériences

3.1 Paramètres et ressources

Nous utilisons dans nos expériences les *plongements* à 300 dimensions pré-entraînés sur Google News avec *word2vec* en les modifiant durant l'entraînement. Les vecteurs de positions et de syntagmes sont de taille 50. La probabilité de *dropout* est fixée à 0,8. Pour chacune des tailles de champ récepteur (2,3,4,5), 150 filtres sont utilisés, pour un total de 600. Ces filtres sont dotés d'une tangente

hyperbolique comme non-linéarité. Le modèle est entraîné par descente de gradient stochastique (SGD) avec l'optimiseur Adadelta et un *clipping* du gradient fixé à 3. La taille des mini-lots est fixée à 50. Le nombre d'époques d'apprentissage est contrôlé par *early stopping* sur le jeu de validation. Les résultats présentés sont des moyennes sur 10 exécutions en utilisant le score micro-f1 de l'outil officiel d'évaluation de TAC 2017. Notre corpus d'entraînement est constitué de l'union des jeux de données DEFT_RICH_ ERE_R2_V2 (LDC2015E68), DEFT_RICH_ ERE_V2 (LDC2015E29) et TAC 2015 (LDC2017E02). Notre corpus de validation est le jeu de données issu de la campagne TAC 2016 (LDC2017E02) et nous nous testons sur les données de la campagne TAC 2017 Event Nugget (LDC2017E02). Il existe au sein de ces jeux de données quelques rares cas de mentions annotées avec plusieurs types d'événements distincts. Parmi ces cas de figure, la grande majorité appartient à l'une des trois combinaisons suivantes : (*Attack/Die*, *Transfer-Money/Transfer-Ownership*, *Attack/Injure*). Pour traiter ce problème, nous introduisons trois types hybrides lors de l'apprentissage pour ces trois types d'événements, ce qui permet de conserver une classification simple (mono-étiquette). Nos jeux de données de validation et de test se focalisent sur les types d'événements les plus difficiles de Rich ERE, à l'instar de TAC 2017, et restreignent la tâche à 19 des 38 types. Nous entraînons toutefois notre modèle sur l'ensemble des 42 classes (classes hybrides et classe nulle incluse) mais nous ignorons les types non présents en test lors de la prédiction. De même, le vecteur global n'agrège que les prédictions des types présents sur le jeu de test. Enfin, différentes normalisations du vecteur de contexte global ont été comparées expérimentalement. Les résultats présentés ici reposent sur la meilleure normalisation obtenue en validation pour chaque configuration : les vecteurs $\mathbf{f}_{\mathbf{[large/phrase]}}$ ne sont pas normalisés alors que le vecteur $\mathbf{f}_{\mathbf{doc}}$ est centré-réduit avant d'être fourni au modèle.

3.2 Influence de la taille du contexte local

Les modèles convolutifs obtenant des performances compétitives sur les bases ACE 2005 (Nguyen & Grishman, 2015), TAC 2016 (Nguyen *et al.*, 2016b) et TAC 2017 (Kodelja *et al.*, 2017) partagent des architectures similaires, notamment concernant la taille du contexte local employé. La fenêtre utilisée est de taille $w = 15$, centrée sur la mention candidate. Afin d'observer la capacité du modèle à exploiter des dépendances longues au sein de la fenêtre, nous présentons à la figure 3 les performances du modèle en fonction de la taille de ce contexte. On constate que les performances du modèle local saturent dès $w = 2$, taille que nous conservons par la suite. Ces résultats indiquent que le modèle convolutif ne parvient pas véritablement à exploiter des dépendances distantes au sein du contexte local. Ce modèle local obtient des performances similaires à celles d'un ensemble de réseaux BiLSTM (voir (Makarov & Clematide, 2017) dans le tableau 2). Il semble donc que cette architecture récurrente, bien que théoriquement mieux à même d'exploiter des dépendances longues, ne le fait en réalité pas mieux qu'un modèle convolutif. Ces résultats motivent l'intérêt théorique de notre approche : les réseaux ne parvenant pas à exploiter plus d'informations lorsque l'on augmente la taille de la fenêtre du contexte local, il est souhaitable de rendre cette information distante accessible. Les sections suivantes se consacrent à l'étude de cette intégration.

3.3 Influence de la taille du contexte global

Comme nous l'avons vu précédemment, l'agrégation du contexte peut se faire à plusieurs échelles. L'agrégation au niveau de la phrase courante peut résoudre les ambiguïtés intra-phrastiques alors qu'une agrégation plus large peut être utile pour la désambiguïsation inter-phrastique. Afin de déterminer le niveau d'agrégation le plus utile, nous comparons dans le tableau 1 l'apport de l'intégration

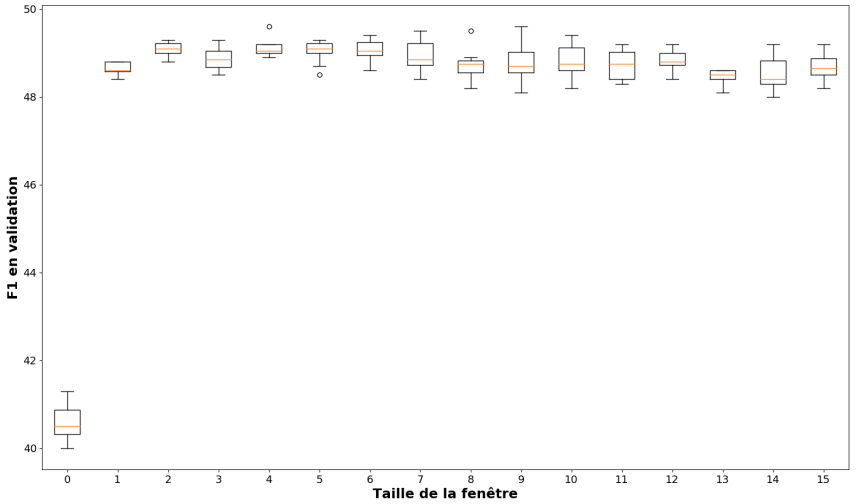


FIGURE 3 – Influence de la taille w du contexte local sur les performances du modèle local en validation. Pour chaque configuration, les résultats sont des moyennes sur 8 entraînements.

méthode	P	R	F
$\text{CNN}_{\text{doc-plongement}}$	**52,71**	47,95	**50,2** ‡
$\text{CNN}_{\text{large-plongement}}$	52	47,6	49,69
$\text{CNN}_{\text{phrase-plongement}}$	49,83	49,49	49,66
$\text{CNN}_{\text{local}}$	46,42	**52,04**	49,06

TABLE 1 – Performances sur la base de validation TAC 2016 en fonction du niveau d'agrégation. Résultats moyennés sur 10 entraînements pour chaque configuration. Seul le modèle $\text{CNN}_{\text{doc-plongement}}$ est significativement meilleur que $\text{CNN}_{\text{local}}$ ($p < 0,01$).

du contexte $\mathbf{f}_{\mathbf{global}}$ à la matrice d'entrée $\mathbf{X}_c$ en fonction du niveau d'agrégation : *phrase*, *large* et *doc*. Nous constatons tout d'abord que les trois niveaux d'agrégation améliorent significativement les performances par rapport à notre baseline $\text{CNN}_{\text{local}}$. De plus, les performances augmentent avec la taille du contexte.

3.4 Comparaison avec l'état de l'art

Afin de valider l'apport de notre méthode, en plus de la comparaison à notre modèle initial $\text{CNN}_{\text{local}}$, nous nous comparons aux 3 modèles ayant obtenus les meilleurs résultats lors de la campagne d'évaluation TAC 2017 :

1. **Méthode d'ensemble BiLSTM CRF** : Jiang *et al.* (2017) utilisent un ensemble de 10 modèles BiLSTM combinés par une stratégie de vote. Mettant en avant le bon rappel des modèles neuronaux au détriment de la précision, ils y adjoignent un classifieur CRF pour améliorer la précision. Pour le BiLSTM, seuls des plongements de mots sont employés, tandis que le CRF

Méthodes	max			moyenne sur 10 exécutions		
	P	R	F	P	R	F
BILSTM CRF (Jiang) †	**56,83**	**55,57**	**56,19**	-	-	-
BILSTM à large marge (Makarov) †	52,16	48,71	50,37	-	-	-
CNN (Kodelja)	54,23	46,59	50,14	-	-	-
CNN_{local}	52,21	49,55	50,84	51,9	48,92	50,36
$CNN_{doc\text{-}plongement}$	**59,13**	45,37	51,34	**58,07**	45,43	50,95 ‡
$CNN_{doc\text{-}softmax}$	52,87	**50,35**	**51,58**	53,12	**49,61**	**51,3** ‡
$CNN_{doc\text{-}plong\_soft}$	55,72	47,08	51,04	57,62	45,09	50,58
$CNN_{doc2vec}$	53,20	47,40	50.10	53,54	46,92	49,98

TABLE 2 – Performance sur la base de test TAC 2017. "†" désigne des modèles d'ensemble. ‡ indique dans la seconde partie du tableau les modèles significativement meilleurs que le modèle CNN_{local} ($p < 0,01$ pour un t-test bilatéral sur les moyennes).

emploie de multiples attributs tels que tokens, lemmes, racines, présence d'entités nommées et étiquettes morphosyntaxiques.

2. **BiLSTM à large marge** : Makarov & Clematide (2017) utilisent un BiLSTM doté d'un objectif à large marge (Gimpel & Smith, 2010). Cet objectif pénalise plus fortement les faux négatifs afin de compenser la rareté des classes positives dans le jeu de données. Un ensemble de 5 réseaux est utilisé pour la prédiction et des types hybrides sont utilisés.
3. **Modèle convolutif** : ce modèle, proposé par (Kodelja *et al.*, 2017) est similaire à notre modèle CNN_{local}. Il s'agit d'un réseau convolutif utilisant des plongements de mots, de positions, de parties du discours et des dépendances syntaxiques en entrée du modèle. La principale différence est l'absence de types hybrides pour gérer les cooccurrences d'événements.
4. **CNN-doc2vec** : à l'instar de Duan *et al.* (2017), nous intégrons un vecteur de document au niveau des plongements de notre modèle. Ce vecteur de taille 100 est généré par le modèle PV-DM (Le & Mikolov, 2014). À la différence de notre représentation globale, celle-ci n'est pas spécifique à la tâche. Nous avons optimisé les mêmes hyperparamètres d'intégration que pour notre représentation, à savoir le choix de la normalisation et du niveau d'intégration. La meilleure configuration présentée ici intègre des vecteurs centrés-réduits au niveau du softmax.

Le tableau 2 présente la comparaison de ces méthodes sur la base de test TAC 2017. Il est difficile de comparer notre contribution à (Jiang *et al.*, 2017) et (Makarov & Clematide, 2017) car ce sont des méthodes d'ensemble alors que nous présentons les performances pour un seul modèle. De plus, leurs scores moyens sur plusieurs initialisations ne sont pas disponibles alors que les variations sont souvent non négligeables (Reimers & Gurevych, 2017).

Le modèle hybride de Jiang *et al.* (2017) n'est en outre pas une méthode d'ensemble simple fondée sur le vote de plusieurs modèles de même architecture mais la combinaison d'un ensemble de BiLSTMs votant pour une prédiction agrégée avec la prédiction d'un CRF selon une heuristique spécifique. Il est à noter de ce point de vue que notre approche ne faisant pas d'hypothèse sur le modèle neuronal de base, il serait théoriquement possible d'intégrer notre représentation globale aux BiLSTMs avant l'application de la stratégie d'ensemble.

Un autre élément rendant les comparaisons difficiles, identifié lors d'analyses récentes, est la pré-

sence de blocs de citations dans les documents. Ces citations ne sont pas annotées en événements, même lorsqu'elles reprennent des phrases précédentes contenant effectivement des événements. Ces phrases constituent donc des duplicatas de phrases contenant possiblement des événements. Durant l'apprentissage, le modèle reçoit alors des annotations contradictoires pour deux exemples pourtant identiques. Durant le test, ignorer ces phrases peut ainsi significativement augmenter les performances. Les résultats présentés ici ne tiennent pas compte de cet aspect, ce qui minore très certainement les performances de notre modèle. Nous avons eu par ailleurs confirmation [2] que les performances rapportées par Makarov & Clematide (2017) négligeaient ce phénomène et que sa prise en compte dans leur modèle entraînait également un gain significatif.

Leur approche peut donc être comparée à la nôtre de ce point de vue [3], comparaison montrant que notre *baseline* $\text{CNN}_{\text{local}}$ est légèrement supérieure en moyenne aux performances du BiLSTM à large marge de Makarov & Clematide (2017). Malgré la tendance actuelle à privilégier les architectures récurrentes, les modèles utilisant la convolution restent donc compétitifs. On peut supposer que de façon similaire à ce que nous avions constaté à la section 3.2 pour les CNNs, les RNNs utilisés n'apprennent pas à exploiter réellement l'intégralité du contexte à disposition, l'influence du contexte proche étant prédominante dans la majorité des cas. Enfin, pour achever la comparaison avec les modèles extérieurs, le tableau 2 montre que l'introduction de types hybrides dans notre baseline $\text{CNN}_{\text{local}}$ permet d'obtenir de meilleurs résultats que ceux du CNN de Kodelja *et al.* (2017).

Concernant plus spécifiquement les modèles proposés, les variantes $\text{CNN}_{\text{doc-plongement}}$ et $\text{CNN}_{\text{doc-softmax}}$ améliorent de manière significative les performances par rapport à notre *baseline* et au modèle d'ensemble de Makarov & Clematide (2017). L'intégration simultanée aux deux niveaux, $\text{CNN}_{\text{doc-plong\_soft}}$, n'obtient pas en revanche de gain significatif. Enfin, on peut observer que l'intégration de la représentation globale proposée par (Duan *et al.*, 2017) provoque une chute des performances. L'absence de spécificité des représentations construites par rapport à la tâche et au corpus considérés est une explication possible de cette contre-performance.

4 Discussions

4.1 Analyse du choix du contexte

En premier lieu, il faut souligner que du point de vue de la taille du contexte à prendre en compte, les résultats de la section 3.2 montrent assez clairement l'intérêt de se situer à l'échelle du document plutôt qu'à une granularité de contexte plus fine. Une interprétation possible de ce constat est que l'intégration d'un contexte global au modèle est intrinsèquement plus adaptée à la résolution des ambiguïtés inter-phrastiques qu'intra-phrastiques. Sur un autre plan, on peut également noter que le contexte global est généré à partir des prédictions d'un premier modèle imparfait. Il est donc bruité. Agréger les prédictions sur un contexte plus large pourrait alors permettre de compenser ce bruit plus efficacement. Afin de distinguer ces deux phénomènes, nous comparons dans le tableau 3 les résultats de l'intégration au niveau de la phrase et du document en utilisant cette fois les annotations réelles à la place des prédictions du $\text{CNN}_{\text{local}}$. On constate que cette fois encore, les meilleures performances sont obtenues en agrégeant l'information à l'échelle du document. L'écart avec $\text{CNN}_{\text{phrase-plongement}}$ est même encore plus élevé. Il semble donc que le niveau d'agrégation ne dépende pas des performances

2. Communication personnelle.
3. Ce que nous ne pouvons pas dire en revanche concernant (Jiang *et al.*, 2017).

méthode	P	R	F
$CNN_{\text{doc-plongement}}$	**54,85**	51,02	**52,83**
$CNN_{\text{phrase-plongement}}$	54,21	47,58	50,68
CNN_{local}	46,42	**52,04**	49,06

TABLE 3 – Performances sur la base de validation TAC 2016 en fonction du niveau d'agrégation avec le contexte parfait. Résultats moyennés sur 10 entraînements pour chaque configuration.

du modèle local et que l'agrégation à l'échelle du document soit intrinsèquement meilleure, peut-être en raison d'une possible prévalence des ambiguïtés inter-phrastiques.

Au-delà de la taille du contexte global, sa nature peut être aussi importante. Dans cet article, ce contexte est issu de l'agrégation des prédictions réalisées à une échelle locale. Une approche alternative serait d'utiliser la représentation issue de la couche précédente du modèle ($\mathbf{f_{pooling}}$), représentation plus riche que ses simples prédictions. Nous avons réalisé des expériences préliminaires concernant cette intégration en faisant varier les mêmes paramètres que dans le reste de l'étude (niveau d'agrégation, normalisation, niveau d'intégration). Néanmoins, ces expérimentations ne se sont pas révélées très concluantes, la meilleure configuration (agrégation à l'échelle du document, intégration au niveau du softmax et représentation centrée réduite) obtenant 50,45 et 50,54 en f1-mesure, respectivement en validation et en test.

Concernant le niveau d'intégration du contexte global, les résultats du tableau 2 montrent que les gains de $CNN_{\text{doc-plong\_soft}}$ et de $CNN_{\text{doc-plongement}}$ sont obtenus en privilégiant la précision au détriment du rappel. Au contraire, la configuration la plus favorable, $CNN_{\text{doc-softmax}}$, permet une amélioration de la précision, certes plus faible, mais ne dégradant pas le rappel. Ce modèle étant le plus favorable, nous nous concentrons sur celui-ci dans le reste de cette étude.

4.2 Analyse d'erreur du meilleur modèle

La figure 4 présente un comparatif des performances du modèle local et du modèle $CNN_{\text{doc-softmax}}$. Pour plus de précision, le tableau 4.2 donne en outre les valeurs correspondantes. Une première observation très clairement visible sur la figure 4 est l'existence d'un écart de performance important entre les différentes classes, aussi bien pour le modèle local que global. Puisque notre modèle global s'appuie sur les prédictions du modèle local, on aurait pu craindre que la représentation globale dégrade les performances sur les classes faibles. On constate que ce n'est pas le cas : sur les 5 classes ayant les performances initiales les plus basses, 2 ne sont pas affectées et les performances augmentent pour une classe. Les deux classes restantes, Contact-Contact et Transaction-Transaction, sont les sous-types utilisés en cas d'ambiguïté avec d'autres sous-types des types Contact et Transaction respectivement. On observe dans les deux cas une amélioration d'une autre classe du même type d'événement (respectivement Contact-Correspondance et Transaction-Transfer-money), ce qui indique un probable transfert entre ces sous-types.

Afin d'illustrer qualitativement l'apport de notre méthode, nous présentons deux exemples de prédiction incorrecte par le modèle local, corrigée par le modèle global. La phrase complète est fournie avec le contexte local entouré de crochets et la mention candidate en gras ainsi que les prédictions du modèle local agrégées à l'échelle du document et l'erreur commise par le modèle local.

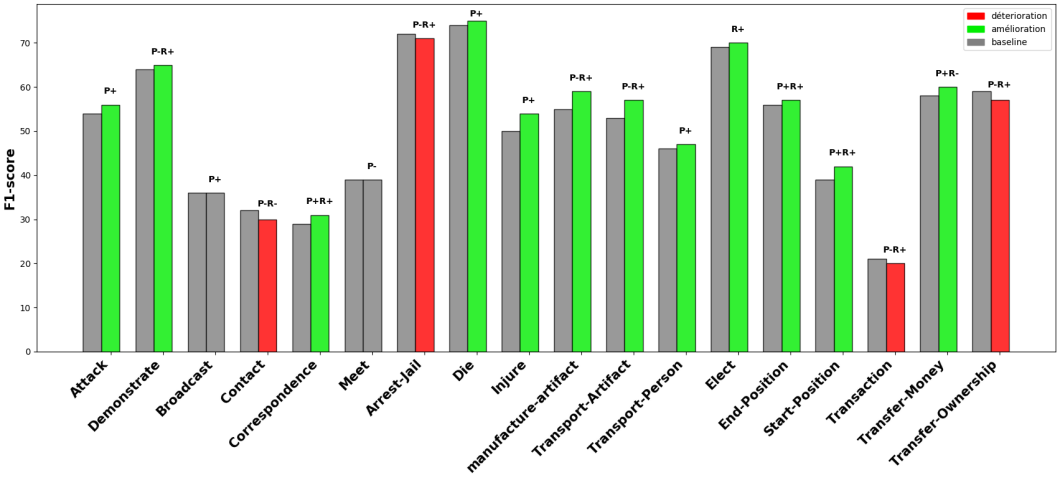


FIGURE 4 – Comparaison des performances par classe entre CNN_{local} et $CNN_{doc\text{-}softmax}$ pour la f1-mesure. Les barres de gauche correspondent au modèle local, celles de droite au modèle global. La barre de droite est verte lorsque l'on observe un gain de f1-mesure pour cette classe, rouge dans le cas contraire et grise en l'absence de variation. Pour les classes présentant une variation, les lettres au dessus indiquent l'origine du changement : **P**récision, **R**appel ou les deux.

— « Do n't get me wrong , I 'm [glad he **won** , I] voted for him . »
Contexte global : (elect : 14, correspondence : 5, contact : 3, transport_person : 3, broadcast : 2)
Faux négatif : elect

— « 100,000 MtGox [Bitcoins were **lost** through theft] (about $ 500 million or 7 % of the outstanding Bitcoins). »
Contexte global : (transfer_ownership : 11, transfer_money : 7, contact_contact : 2, transport_person : 2, manufacture_artifact : 2, die : 1, transaction : 1, arrest_jail : 1)
Faux positif : die

Dans le premier exemple, le contexte local ne permet pas d'identifier clairement que *won* fait référence à un événement de la classe *elect*. Le modèle local ayant détecté de nombreuses autres mentions plus évidentes appartenant à cette classe, le modèle parvient global parvient à identifier le type de la mention. Dans le deuxième exemple, à l'inverse, le modèle local interprète incorrectement *lost* comme faisant référence à un décès. Cependant, les documents faisant référence à des décès contiennent généralement plusieurs mentions de ce type ou d'autres types connexes (*injure*, *attack*). Grâce à cette information notre modèle global ne prédit plus incorrectement cette classe. On notera par ailleurs que dans ce second exemple, même avec un contexte local restreint, il apparaît comme évident que l'on a pas affaire à un événement de type *die*. Ceci met en lumière la forte sensibilité des modèles neuronaux vis-à-vis de la mention candidate, au détriment de son contexte.

Conclusion et perspectives

Dans cet article, nous proposons une nouvelle méthode de représentation du contexte global pour la tâche d'extraction d'événements. Cette méthode est fondée sur l'amorçage. Elle agrège à l'échelle

Type	CNN_{local}			$CNN_{doc\text{-}softmax}$		
	P	R	F	P	R	F
conflict-attack	49	61	54	**52**	61	**56**
conflict-demonstrate	62	66	64	61	**69**	**65**
contact-broadcast	59	26	36	**60**	26	36
contact-contact	25	43	32	24	39	30
contact-correspondence	34	25	29	**38**	**27**	**31**
contact-meet	49	33	39	47	33	39
justice-arrest_jail	63	84	72	62	**85**	71
life-die	70	78	74	**72**	78	**75**
life-injure	47	55	50	**53**	55	**54**
manufacture-artifact	66	47	55	59	**59**	**59**
movement-transport_artifact	75	41	53	74	**46**	**57**
movement-transport_person	43	50	46	**44**	50	**47**
personnel-elect	64	74	69	64	**76**	**70**
personnel-end_position	67	47	56	**68**	**48**	**57**
personnel-start_position	41	38	39	**46**	**39**	**42**
transaction-transaction	32	16	21	24	**18**	20
transaction-transfer_money	54	64	58	**59**	61	**60**
transaction-transfer_ownership	67	53	59	60	**54**	57

TABLE 4 – Comparaison détaillée des performances par classe entre CNN_{local} et $CNN_{doc\text{-}softmax}$. Pour une meilleure visibilité, nous rapportons les mesures sans les décimales. Les performances en **P**récision, **R**appel et **F**-score sont en gras lorsque le modèle global est meilleur que le modèle local, le nom de la classe l'est seulement quand le F-score est meilleur.

du document les prédictions d'un premier modèle local de nature convolutive, obtenant ainsi une représentation du document focalisée sur la tâche finale. Cette représentation est ensuite intégrée à un nouveau CNN. Nous obtenons ainsi des gains significatifs par rapport au modèle local et des performances supérieures, avec un seul modèle, à une association de modèles BiLSTMs.

Notre représentation globale actuelle n'agrège que les prédictions de la couche de sortie du modèle local. Elle souffre donc des imperfections de ce modèle. Pour dépasser cette limite, nous envisageons d'étudier la génération et l'intégration de représentations plus riches tout en restant spécifiques à la tâche. Ces représentations, de nature hiérarchique, pourraient notamment être fondées sur des modèles de classification thématique ou de clustering semi-supervisé.

Remerciements

Ce travail a été partiellement financé par l'Agence Nationale de la Recherche dans le cadre du projet ANR-15-CE23-0018 ASRAEL.

Références

CHEN Y., XU L., LIU K., ZENG D. & ZHAO J. (2015). Event Extraction via Dynamic Multi-Pooling Convolutional Neural Networks. In *53rd Annual Meeting of the Association for Computational Linguistics and 7th International Joint Conference on Natural Language Proce ssing (ACL-IJCNLP 2015)*, p. 167–176, Beijing, China.

DUAN S., HE R. & ZHAO W. (2017). Exploiting Document Level Information to Improve Event Detection via Recurrent Neural Networks. In *Eighth International Joint Conference on Natural Language Processing (IJCNLP 2017)*, p. 352–361, Taipei, Taiwan.

FENG X., HUANG L., TANG D., JI H., QIN B. & LIU T. (2016). A Language-Independent Neural Network for Event Detection. In *54th Annual Meeting of the Association for Computational Linguistics (ACL 2016)*, p. 66–71, Berlin, Germany.

GIMPEL K. & SMITH N. (2010). Softmax-Margin CRFs: Training Log-Linear Models with Cost Functions. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics (NAACL HLT 2010)*, p. 733–736, Los Angeles, California.

JIANG S., LI Y., QIN T., MENG Q. & DONG B. (2017). SRCB Entity Discovery and Linking (EDL) and Event Nugget Systems for TAC 2017. In *Text Analysis Conference (TAC)*.

KODELJA D., BESANÇON R., FERRET O., LE BORGNE H. & BOROS E. (2017). CEA LIST Participation to the TAC 2017 Event Nugget Track. In *Text Analysis Conference (TAC)*.

LE Q. & MIKOLOV T. (2014). Distributed Representations of Sentences and Documents. In *31st International Conference on International Conference on Machine Learning (ICML 2014)*, p. 1188–1196, Beijing, China.

MAKAROV P. & CLEMATIDE S. (2017). UZH at TAC KBP 2017: Event Nugget Detection via Joint Learning with Softmax-Margin Objective. In *Text Analysis Conference (TAC)*.

MANNING C. D., SURDEANU M., BAUER J., FINKEL J., BETHARD S. J. & MCCLOSKY D. (2014). The Stanford CoreNLP Natural Language Processing Toolkit. In *52nd Annual Meeting of the Association for Computational Linguistics (ACL 2014), system demonstrations*, p. 55–60.

MIKOLOV T., SUTSKEVER I., CHEN K., CORRADO G. S. & DEAN J. (2013). Distributed Representations of Words and Phrases and Their Compositionality. In *26th International Conference on Neural Information Processing Systems (NIPS 2013)*, p. 3111–3119, Lake Tahoe, Nevada.

NGUYEN T. H., CHO K. & GRISHMAN R. (2016a). Joint Event Extraction via Recurrent Neural Networks. In *2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL HLT 2016)*, p. 300–309, San Diego, California.

NGUYEN T. H. & GRISHMAN R. (2015). Event Extraction via Dynamic Multi-Pooling Convolutional Neural Networks. In *53rd Annual Meeting of the Association for Computational Linguistics and 7th International Joint Conference on Natural Language Proce ssing (ACL-IJCNLP 2015)*, p. 365–371, Beijing, China.

NGUYEN T. H., GRISHMAN R. & MEYERS A. (2016b). New York University 2016 System for KBP Event Nugget: A Deep Learning Approach. In *Text Analysis Conference (TAC)*.

REIMERS N. & GUREVYCH I. (2017). Reporting Score Distributions Makes a Difference: Performance Study of LSTM-networks for Sequence Tagging. In *2017 Conference on Empirical Methods in Natural Language Processing (EMNLP 2017)*, p. 338–348, Copenhagen, Denmark.

SONG Z., BIES A., STRASSEL S., RIESE T., MOTT J., ELLIS J., WRIGHT J., KULICK S., RYANT N. & MA X. (2015). From Light to Rich ERE: Annotation of Entities, Relations, and Events. In *3rd Workshop on EVENTS: Definition, Detection, Coreference, and Representation*, p. 89–98, Denver, Colorado.

Construction conjointe d'un corpus et d'un classifieur pour les registres de langue en français

Gwénolé Lecorvé[1] Hugo Ayats[1] Benoît Fournier[1] Jade Mekki[1, 2]
Jonathan Chevelu[1] Delphine Battistelli[2] Nicolas Béchet[3]
(1) Univ Rennes, CNRS, IRISA, 6, rue de Kerampont, 22305 Lannion Cedex, France
(2) Universite Paris-Ouest-Nanterre, MoDyCo, 200, avenue de la République 92001 Nanterre Cedex, France
(3) Universite de Bretagne Sud, IRISA, Campus de Tohannic, rue Yves Mainguy, 56017 Vannes Cedex, France
prenom.nom@irisa.fr, delphine.battistelli@u-paris10.fr

RÉSUMÉ

Les registres de langue sont un trait stylistique marquant dans l'appréciation d'un texte ou d'un discours. Cependant, il sont encore peu étudiés en traitement automatique des langues. Dans cet article, nous présentons une approche semi-supervisée permettant la construction conjointe d'un corpus de textes étiquetés en registres et d'un classifieur associé. Cette approche s'appuie sur un ensemble initial et restreint de données expertes. Via une collecte automatique et massive de pages web, l'approche procède par itérations en alternant l'apprentissage d'un classifieur intermédiaire et l'annotation de nouveaux textes pour augmenter le corpus étiqueté. Nous appliquons cette approche aux registres familier, courant et soutenu. À l'issue du processus de construction, le corpus étiqueté regroupe 800 000 textes et le classifieur, un réseau de neurones, présente un taux de bonne classification de 87 %.

ABSTRACT

Joint building of a corpus and a classifier for language registers in French.

Language registers are an observable stylistic trait of texts and speeches. However, they are still poorly studied in natural language processing. In this paper, we present a semi-supervised approach which jointly builds a corpus of texts labeled in registers and an associated classifier. This approach is based on an initial and limited set of expert data. Using an massive automatically retrieved collection of web pages, it iteratively proceeds by alternating the learning of an intermediate classifier and the annotation of new texts to augment the labeled corpus. We apply this approach to formal, neutral, and informal registers. At the end of the process, the labeled corpus gathers 800, 000 texts, and the classifier, a neural network, has an accuracy of 87 %.

MOTS-CLÉS : Registres de langue, apprentissage semi-supervisé, construction de corpus, classification automatique.

KEYWORDS: Language registers, classification, semi-supervised learning, corpus building.

1 Introduction

Les registres de langue fournissent de nombreuses informations sur un locuteur et sa relation avec les destinataires du message. Il s'agit d'un sujet cependant encore peu étudié en traitement automatique des langues (TAL), notamment en raison du manque de données d'apprentissage. Pour pallier ce

problème, cet article présente une approche semi-supervisée de construction d'un corpus textuel étiqueté en registres de langue.

L'approche proposée s'appuie sur un ensemble restreint de données manuellement étiquetées et une vaste collection de pages web automatiquement collectées mais non étiquetées. Le principe de construction tient alors dans l'apprentissage conjoint et itératif d'un classifieur, un réseau de neurones, sur les données étiquetées. Pour une itération donnée, le classifieur permet de catégoriser les données web, de sélectionner celles dont la classification semble fiable, puis de raffiner le classifieur sur la base de l'ensemble des données étiquetées augmenté de celles sélectionnées. Par ce procédé, nous visons une convergence de l'apprentissage du classifieur et de la construction du corpus vers un compromis entre taux de bonne classification et taille du corpus étiqueté. En pratique, nous appliquons ce processus sur un ensemble de $400\,000$ pages web et obtenons un corpus étiqueté en registres familier, courant et soutenu d'environ 750 millions de mots, ainsi qu'un réseau de neurones avec un taux de classification de $87\,\%$. Le jeu de descripteurs utilisé regroupe 46 caractéristiques de natures variées (lexicales, morphologiques, syntaxiques…) issues d'une analyse experte préalable.

Dans cet article, nous présentons tout d'abord en section 2 un état de l'art lié aux registres de langue et à leur traitement en TAL. Les sections 3, 4 et 5 introduisent ensuite les détails respectifs de notre approche, des données utilisées et du classifieur. Enfin, les résultats sont présentés en section 6.

2 État de l'art et positionnement

La notion de registre renvoie à la manière dont les productions linguistiques sont évaluées et catégorisées au sein d'une même communauté linguistique (celle du français par exemple) (Ure, 1982; Biber & Conrad, 2009). C'est ainsi que l'on distingue différents registres caractérisés selon de multiples traits spécifiques (termes plus ou moins complexes, ordre des mots, temps des verbes, longueur des phrases…) et souvent considérés sur une échelle de niveaux (par exemple, soutenu, littéraire, courant, familier, populaire, vulgaire…). Le partitionnement en catégories peut couvrir différents spectres selon la définition retenue de registre – le terme « registre » étant lui-même source de discussion – et traduire des finesses d'analyse variables (Sanders, 1993; Biber & Finegan, 1994; Gadet, 1996). Le sujet peut ainsi recouvrir, par exemple, l'influence du média de communication (Charaudeau, 1997) ou du degré de spécialisation (Borzeix & Fraenkel, 2005; Moirand, 2007) sur le discours. Dans notre travail, nous adoptons une vision plus traditionnelle avec un découpage en 3 registres : familier, courant et soutenu. Ce choix est avant tout motivé par le pragmatisme, ce découpage étant en effet relativement consensuel et peu sujet à ambiguïté pour l'étiquetage manuel d'un ensemble de données initial, tout en n'interdisant pas d'éventuels raffinements pour l'avenir. À défaut de caractérisation expérimentale détaillée – puisque c'est précisément l'objectif du projet dans lequel s'inscrit ce travail, nos 3 registres considérés se définissent par contraste vis-à-vis d'un emploi central, neutre, de la langue, c'est-à-dire la langue telle qu'employée lorsque le destinataire du message n'est pas connu. Pour assurer la bonne compréhension de ce message, cet emploi implique un ensemble minimal d'hypothèses quant aux connaissances spécifiques du destinataire et se calque donc sur la grammaire et le vocabulaire de la langue, sans pour autant en exploiter les tournures ou termes les plus rares. Ce périmètre d'usage définit le registre courant. Le registre soutenu peut alors être considéré comme l'ajout d'une hypothèse sur un haut degré de maîtrise de la langue de la part du lecteur ou interlocuteur. À l'inverse, le registre familier relâche les contraintes de respect de la norme en autorisant des écarts (volontaires ou fautifs) à différents niveaux (grammaire, vocabulaire mais aussi orthographe,

typographie…). Le registre familier fait alors également l'hypothèse d'une certaine compréhension de ces écarts comme autant de codes spécifiques. C'est à travers cette notion récurrente de connaissances partagées, et de donc de communauté, que les registres de langue s'enracinent dans le domaine de la sociolinguistique. Nous n'intégrons cependant pas cette dimension dans cet article.

À notre connaissance, les registres ont été peu étudiés en TAL, voire pas du tout sous l'angle que nous adoptons. Pour autant, tout un pan de travaux s'intéresse à l'utilisation du langage dans des situations particulières, cherchant à caractériser des « sous-langages », à les identifier, à les classer ou à les imiter. Ces sous-langages peuvent être portés par les notions de thème, de type documentaire, de style phonologique, de polarité en termes d'opinion, d'émotion… À notre connaissance cependant, aucun travail ne s'intéresse à la notion de registres de langue mais beaucoup de travaux en traitement de style apportent une base solide en termes de méthodologie et d'outils théoriques. Sans être exhaustif, l'étude des registres de langue partagent des similitudes avec ceux en attribution d'auteur (Stamatatos, 2009; Iqbal *et al.*, 2013), analyse des nouveaux médias (Schler *et al.*, 2006; Kobus *et al.*, 2008; Gianfortoni *et al.*, 2011; Eisenstein, 2013; Cougnon & Fairon, 2014). Différents corpus de référence ont d'ailleurs été publiés pour ces différents médias. Notre travail vise à combler le manque d'équivalent pour la notion de registre.

Les méthodes de traitement de style automatique sont toutes fondées sur un ensemble de descripteurs pertinents dérivés des textes à traiter. En raison de son importance historique, le travaux en attribution d'auteur permettent d'identifier un large éventail de descripteurs. Comme l'indique Stamatatos (2009), les préférences ou les choix d'écriture d'un auteur sont reflétés à plusieurs niveaux de langage. Le plus évident et le plus étudié est le niveau lexical, par exemple à travers la longueur des mots et des phrases d'un texte, la richesse de son vocabulaire ou les fréquences des mots et des n-grammes de mots (De Vel *et al.*, 2001; Sanderson & Guenter, 2006). À cet égard, il est généralement admis dans la communauté que les mots-outils (prépositions, articles, auxiliaires, verbes modaux…) sont d'intérêt notable alors que d'autres mots (noms, adjectifs…) doivent être évités pour le traitement de la style (Koppel & Schler, 2003; Argamon *et al.*, 2007), selon un principe d'orthogonalité entre le style et la signification d'un texte. Ce principe souligne l'importance d'abstraire certains éléments de sens pour l'analyse du style, faute de quoi l'analyse risque d'être biaisée par le thème des textes traités. Malgré tout, quelques descripteurs sémantiques peuvent se révêler utiles, par exemple les fréquences de recours à des synonymes et hyperonymes ou les relations fonctionnelles entre propositions (clarification d'une proposition par une autre, mise en oppositiondots) (McCarthy *et al.*, 2006; Argamon *et al.*, 2007). Par ailleurs, quelque soit leur sens, l'emploi de certains mots témoigne explicitement de l'appartenance du texte à un style précis (Tambouratzis *et al.*, 2004), en particulier dans le cas des registres de langue. Sur le plan syntaxique, l'emploi de descripteurs issus d'analyses morphosyntaxiques et syntaxiques est très largement répandu pour caractériser le style (Koppel & Schler, 2003; Hirst & Feiguina, 2007; Sidorov *et al.*, 2014). Enfin, d'autres travaux se sont intéressés à l'information graphémique en considérant des n-grammes de caractères, les types des graphèmes (lettre, chiffre, ponctuation, majusculedots) ou encore des mesures de compression de l'information (Koppel & Schler, 2003; Marton *et al.*, 2005; Escalante *et al.*, 2011). Dans notre travail, une étude linguistique préliminaire a été menée en ce sens (Mekki *et al.*, 2017, 2018), conduisant à un ensemble de descripteurs pour les 3 registres considérés.

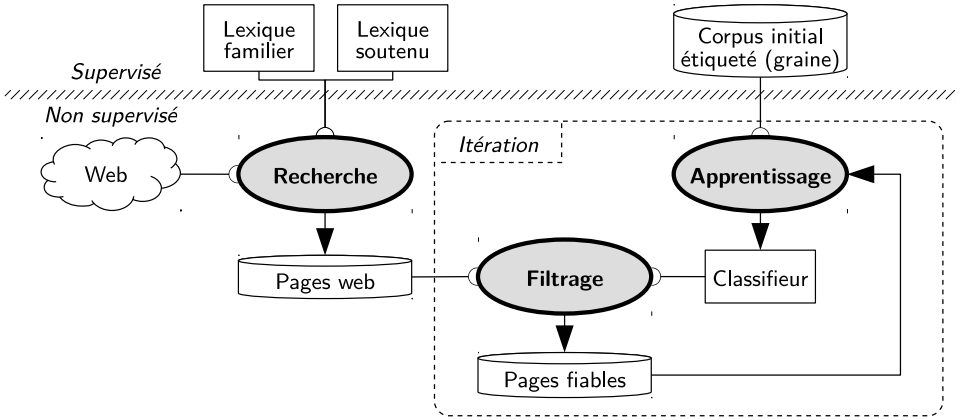


FIGURE 1 – Vue d'ensemble du processus semi-supervisé.

3 Approche proposée

Cette section décrit le processus semi-supervisé de construction d'un corpus étiqueté en registre. Comme illustré sur la figure 1, le processus est amorcé par une étape de collecte de données sur Internet. Cette collecte s'appuie sur deux lexiques spécialisés, l'un pour le registre familier, l'autre pour le soutenu, à partir desquels des requêtes familières ou soutenues sont formées, puis soumises à un moteur de recherche. Après nettoyage automatique, les pages récupérées sont regroupées au sein d'un unique corpus dont on cherche à extraire les plus pertinentes pour chaque registre. Cette extraction se fait par le biais d'un classifieur probabiliste (un réseau de neurones) prédisant la probabilité d'appartenance à chaque registre. Pour résoudre l'interdépendance selon laquelle le classifieur nécessite des données d'entraînement étiquetées et l'étiquetage des données nécessite un classifieur, l'approche procède par itérations. Ainsi, un premier classifieur est initialement entraîné sur une graine, c'est-à-dire un faible ensemble initial de données annotées manuellement et indépendant des pages web récupérés. Ce classifieur permet de sélectionner les textes dont l'appartenance à l'un des registres est considérée comme fiable, dans notre cas si la probabilité d'appartenance à un registre est supérieure à un seuil donné. Ces textes sont ensuite ajoutés à ceux déjà étiquetés, puis une nouvelle itération démarre. Ce processus semi-supervisé permet en fin de processus d'obtenir conjointement un ensemble de textes catégorisés et un classifieur. Notons que le recours à Internet n'est pas une originalité de notre travail puisque de nombreux exemples analogues existent dans littérature, par exemple (Baroni & Bernardini, 2004) (bien que notre processus de collecte ne soit pas itératif ici) ou encore (Lecorvé *et al.*, 2008) pour la collecte de pages thématiques.

Les classes considérées sont « familier », « soutenu » et « courant ». La considération du registre courant se justifie par le fait que les deux premiers registres se définissent par leurs variations respectives à ce troisième. Ainsi, le registre courant, parfois qualifié de neutre, rassemble les textes qui présentent peu d'écart à la norme. Pour compléter ce partitionnement des textes, nous faisons également l'hypothèse que certains textes récupérés n'appartiennent à aucun des 3 registres, soit car le texte est mal formé (langue étrangère, style SMS, texte non naturel…), soit car le registre n'y est pas homogène (par exemple, dans une liste de commentaires). Notre condition de fiabilité d'appartenance permet de modéliser cela.

Requêtes familières	Requêtes soutenues
Exemples positifs	
nain porte quoi	couronne de myrte
roublardise foutre la paix	dioscurisme argutieusement
croquenot se cuiter	hic et nunc
avaler sa chique	géronte séductible
Exemples négatifs	
montrer le chemin	relation sexuelle

TABLE 1 – Exemples de requêtes issues des lexiques familier et soutenu.

4 Données

En pratique, les lexiques sur lesquels s'appuie la collecte de pages web sont constitués de mots et expressions automatiquement récupérés à partir d'une sauvegarde de la version française de Wiktionary [1]. Pour un registre donné, seuls les mots sans ambiguïté d'appartenance à un registre sont considérés, c'est-à-dire les termes ayant toutes leurs acceptions annotées comme appartenant à un même registre. Précisément, les termes annotés comme argotiques, familiers, populaires et vulgaires ont été regroupés au sein du lexique familier et ceux catégorisés comme littéraires et soutenus au sein du lexique soutenu, chacun totalisant ainsi respectivement $6\,000$ et 500 entrées. Les requêtes sont construites registre par registre en combinant au hasard des éléments choisis du lexique associé. Le nombre de requêtes ainsi formées pour chaque lexique est identique afin d'aboutir à un ensemble des pages récupérées sensément équilibré en terme de registre. La longueur des requêtes est empiriquement limitée à un minimum de 2 mots et un maximum de 6 mots afin de garantir une pertinence minimale pour les pages retournées et un nombre de résultats non nul. Les requêtes web sont effectuée à l'aide de l'API Bing [2]. Au total, $12\,000$ requêtes sont soumises, chacune conduisant à un maximum de 50 pages Web. Quelques exemples de ces requêtes sont listés dans la table 1. Bien que certaines requêtes ne fassent *a priori* pas sens (par exemple, « croquenot se cuiter »), $76\,\%$ des requêtes renvoient au moins un résultat et $49\,\%$ en renvoient plus de 50, ces pourcentages étant comparables pour les requêtes familières et soutenues. Par ailleurs, la proportion de requêtes associées à tort à un registre (exemples négatifs dans table 1) est minime. Ces exemples sont en partie dus au fait que quiconque peut éditer Wiktionary, y compris des non-spécialistes. Enfin, signalons que certains dictionnaires en ligne apparaissant régulièrement dans les listes de résultats ont été exclus au moment de la requête afin de ne récolter que des pages où les termes recherchés sont bien en contexte et non isolées dans une définition ou un exemple.

Le contenu textuel des pages web est extrait automatiquement grâce à un outil de nettoyage [3]. Cet outil cherche le corps textuel de la page et ne s'intéresse qu'aux portions de texte « rédigées ». Il exclut ainsi les titres, menus, mentions légales, annonces, etc. mais inclut les commentaires si ceux-ci ont suffisamment de matière linguistique et se conforment au style rédactionnel normé (ponctuation, non abréviation des mots…). Enfin, pour éviter un manque d'homogénéité au sein de pages web longues (par exemple des forums) et de ne pas introduire de biais d'apprentissage liés aux disparités

1. http://fr.wiktionary.org
2. https://docs.microsoft.com/en-us/rest/api/cognitiveservices/bing-web-api-v5-reference
3. http://github.com/glecorve/web-cleaner

de longueur de textes, les textes nettoyés ont été segmentés sur les frontières de paragraphes de sorte à obtenir des segments de 5 000 caractères environ. À partir d'un total de 400 000 pages web et après filtrage des pages n'étant pas en français, le corpus de départ pour nos itérations consiste en environ 825 000 segments textuels, représentant 750 millions de mots. Un effet de ce découpage est d'atténuer l'hypothèse selon laquelle tous les textes contiennent au moins un terme très marqué en matière de registre. Cela apporte de la diversité au corpus mais pourrait également empêcher d'apprendre certaines corrélations entre ces indicateurs saillants et d'autres potentiellement plus discrets.

Enfin, notre ensemble de textes manuellement étiquetés rassemblent des segments issus de romans [4], journaux [5] et pages web. Ces pages web ne proviennent pas de l'ensemble collectés automatiquement pour la construction du corpus et elles ne contiennent ainsi pas nécessairement de termes listés dans nos lexiques spécialisés. Ce constat s'applique également aux textes provenant d'autres sources. L'étiquetage des pages s'est fait par 2 annotateurs sur la base des éléments de définition et de caractérisation relevés par notre étude linguistique préalable (différences entre registres, traits linguistiques à observer, exemples…). Au total, 435 segments textuels, soit environ 440 000 mots, sont considérés, équitablement répartis entre les registres familier, courant et soutenu.

5 Apprentissage du classifieur

Le classifieur s'appuie sur un ensemble de 46 caractéristiques listées par la table 2 et extraites automatiquement à partir de chaque texte. Celles-ci sont issue d'une expertise linguistique préliminaire (étude de l'état de l'art et analyse en corpus) dont les détails peuvent être trouvés dans (Mekki *et al.*, 2017) et (Mekki *et al.*, 2018). Elles couvrent de multiples niveaux d'abstraction de la langue, y compris des aspects liés à l'oral car le registre familier partagent des liens avec cette pratique de la langue (retranscription de certaines élisions de phonèmes, allongement de certaines syllabes…). Ces descripteurs sont tous des fréquences relatives globales à chaque texte (par exemple, le nombre de mots avec des répétitions de voyelles rapporté au nombre de mot dans le texte). Les ressources utilisées pour les descripteurs lexicaux ont été extraites de Wiktionnaire. Les analyses orthographiques et grammaticales (morphosyntaxe et syntaxe) ont été produites grâce à l'outil LangageTool [6]. Le reste du travail est réalisé par un ensemble de scripts Python *ad hoc*.

Diverses remarques sont à formuler concernant l'appartenance de certains mots ou expressions (plusieurs mots) au lexique d'un registre particulier. Tout d'abord, notons qu'aucun lexique du registre courant n'existe. Ensuite, certains mots peuvent être ambigus quant à leur appartenance à un registre, en fonction de leur contexte d'usage. Par exemple, le mot « caisse » peut, certes, faire référence à une voiture en argot mais il peut également porter le simple sens d'un contenant. Ainsi, deux variantes de descripteurs sont considérés pour les fréquences de mots propres à un registre. La première pondère la fréquence d'un mot par le nombre d'acceptions identifiées comme appartenant au registre considéré divisée par le nombre total de ses acceptions. Dans notre exemple, l'observation du mot « caisse » ne compter que pour moitié. L'autre variante est plus stricte. Elle ne comptabilise un mot que si toutes ses acceptions sont identifiées comme appartenant au registre. Le cas des expressions ne nécessite pas cette dualité car celles-ci sont généralement moins ambiguës. Enfin, nous soulignons que la richesse lexicale du registre familier est bien plus grande que celle du registre soutenu. Il s'agit d'un

4. Parmi lesquels Kiffe kiffe demain de Faïza Guène, Albertine disparue de Marcel Proust, Les Mohicans de Paris d'Alexandre Dumas, Les bâtiments de ponts de Rudyard Kipling, Les misérables de Victor Hugo…

5. Une sélection d'articles de L'Humanité.

6. `https://languagetool.org/`

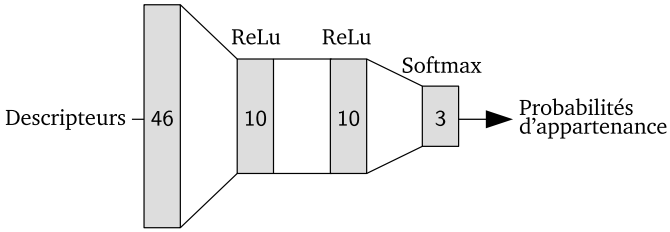


FIGURE 2 – Architecture du réseau de neurones.

phénomème bien connu ayant trait au fait qu'il n'existe qu'une norme du langage mais une infinité de s'en écarter. La richesse du registre familier reflète ces multiples écarts possibles.

Le classifieur est un réseau de neurones multi-couches. Le choix de cet outil d'apprentissage automatique n'est pas une revendication de notre travail. Ce choix se justifie avant tout par la facilité actuelle à construire des réseaux de neurones grâce aux multiples boîtes à outils disponibles. Par ailleurs, les possibilités d'interconnexions entre neurones et les multiples fonctions d'activation existantes permettent de modéliser par des réseaux de neurones d'autres techniques comme des classifieurs naïfs de Bayes ou des modèles de type exponentiel. Enfin, les réseaux de neurones sont connus pour être liés à la propriété de prolongement (Mikolov *et al.*, 2013; Le & Mikolov, 2014). Bien que l'objectif du présent article soit la construction d'un corpus et d'un premier classifieur. Des perspectives futures pourraient être d'observer les similarités entre documents tels présentes dans l'espaces des embeddings produits par notre modèle. Tel que l'illustre la figure 2, le réseau de neurones que nous considérons prend en entrée le vecteur des 46 valeurs représentant un texte. Les valeurs en sortie sont les probabilités d'appartenance à chaque registre. Toutes les couches du réseau sont des couches denses. Les 2 premières sont composées de 10 neurones, la première avec une fonction d'activation de type *leaky ReLU* [7], l'autre avec la fonction *tanh*. La dernière couche est composée de 3 neurones avec une fonction *softmax* afin de produire une distribution de probabilités. Cette architecture est issue de quelques tests sur un ensemble de développement mais n'a pas fait l'objet d'une étude approfondie. Une fois le réseau appris, les probabilités d'appartenance pour un texte fourni en entrée sont directement interprétées comme le niveau de confiance du réseau. Un seuil est alors appliqué pour déterminer s'il faut classer le texte ou non.

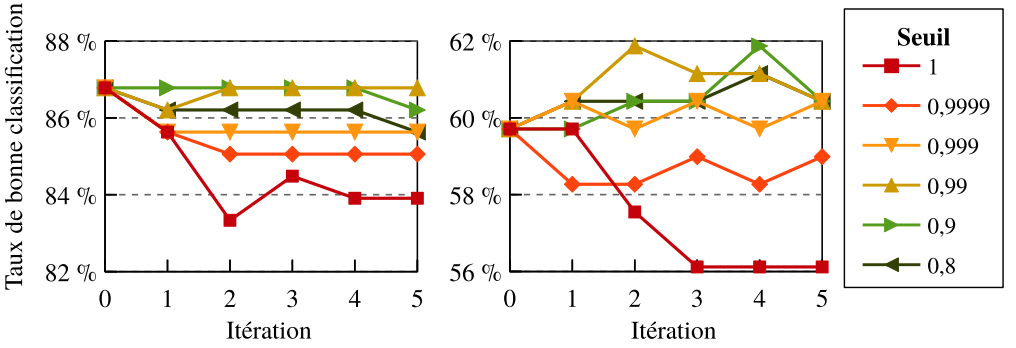
6 Résultats

Les expériences ont été menées en utilisant les bibliothèques Keras [8] et TensorFlow [9]. Hormis lors de l'apprentissage du premier modèle sur la graine, les classifieurs successifs sont appris par lot de 100 instances sur 20 époques en utilisant l'algorithme d'optimisation *rmsprop* et l'erreur absolue moyenne comme fonction objectif. Les 435 segments initialement annotés sont répartis en un ensemble d'apprentissage (40%, soit 174 segments), de développement (20%) et de test (40%). À chaque itération, les segments nouvellement sélectionnés parmi les données web sont injectés dans l'ensemble d'apprentissage pour 80% et l'ensemble de développement pour le reste. L'ensemble de test n'est jamais modifié afin de pouvoir mesurer l'évolution du classifieur tout au long du processus.

7. Paramètre α fixé à $0,1$.
8. `https://keras.io/`
9. `https://www.tensorflow.org/`

Lexique
– Mots familiers pondérés par leur nombre d'acceptions familières : 7 828 éléments
– Mots soutenus pondérés par leur nombre d'acceptions soutenues : 565 éléments
– Mots strictement familiers (toutes les acceptions sont familières) : 3 075 éléments
– Mots strictement soutenus (toutes les acceptions sont soutenues) : 166 éléments
– Expressions familières : 3 453 éléments
– Expressions soutenues : 143 éléments
– Noms d'animaux : 78 éléments
– Onomatopées (« ah », « pff »…) : 125 éléments
– Termes du langage SMS (« slt », « lol », « tkt »…) : 540 éléments
– Anglicisme (lexique et syntaxe)
– Mots inconnus
– Emploi de « ça »
– Emploi de « ce »
– Emploi de « cela »
– Emploi de « des fois »
– Emploi de « là »
– Emploi de « parfois »
Phonétique
– Élision voyelle (« m'dame », « p'tit »…)
– Élision « r » (« vot' », « céleb' »…)
– Liaisons écrites « z » (« les zanimaux »)
Morphologie
– Répétitions de syllabes (« baba », « dodo »…)
– Répétitions de voyelles (« saluuuut »)
– Emploi de mots terminant en « -asse »
– Emploi de mots terminant en « -iotte »
– Emploi de mots terminant en « -o »
– Emploi de mots terminant en « -ou »
– Emploi de mots terminant en « -ouze »
Morphosyntaxe
– Emplois des temps : impératif présent, indicatif futur, indicatif imparfait, indicatif passé simple, indicatif présent, conditionnel présent, subjonctif imparfait, subjonctif présent
– Emploi des personnes : seconde pluriel (« vous ... »), seconde singulier (« tu ... »)
– Emploi de verbe du premier ou deuxième groupes
Syntaxe
– Redoublement de la possession (« son … à lui »)
– Structure « c'est ... qui »
– Emploi de « est-ce que »
– Emploi de la conjonction « et »
– Négations sans « ne »
– Autres fautes de syntaxe

TABLE 2 – Liste des descripteurs utilisés par le classifieur



(a) Test **(b) Pages collectées annotées**

FIGURE 3 – Taux de bonne classification pour chaque itération sur le test (a) et le sous-ensemble annoté des pages récupérées (b).

Par ailleurs, un sous-ensemble des 139 pages web collectées a été tiré aléatoirement et annoté manuellement. Ces pages proviennent équitablement des requêtes familières et soutenues. Parmi ces pages, 27 sont étiquetées pour le registre familier (19 % des pages), 69 pour le registre courant (50 %), 38 pour le registre soutenu (27 %) et 5 comme non étiquetables (4 %) car équivoques [10]. Ce deuxième ensemble est complémentaire de l'ensemble de test car il est constitué de pages qui contiennent des mots connus comme appartenant à un registre, hypothèse absente pour l'ensemble de test. De plus, nous pouvons dores et déjà souligner que les proportions respectives de chaque registre diffèrent entre l'ensemble de test et le sous-ensemble annoté des pages web collectées.

Nous étudions tout d'abord les résultats du classifieur, puis le corpus produit en parallèle.

6.1 Classifieur

La figure 3 montre l'évolution du taux de bonne classification du modèle au fur et à mesure des itérations sur l'ensemble de test (a) et sur le sous-ensemble annoté des pages collectés. Les résultats sont présentés pour différentes valeurs du seuil de sélection des pages, allant de $0,8$ (probabilité d'appartenance) à 1 (c.-à-d. que le classifieur est sûr de lui). Ces valeurs élevées se justifient par le taux élevé de bonne classification de 87 % dès l'initialisation du processus. Sur l'ensemble de test, nous pouvons constater que le classifieur est très stable en dépit des apports en nouvelles données, quelque soit l'ensemble de données. Cela semble signifier que ces nouvelles données sont cohérentes avec notre graine mais qu'elle n'apporte pas d'éléments supplémentaires permettant d'améliorer les performances. Parmi les seuils testés, la consigne de sélection la plus stricte (seuil = 1) conduit à une nette dégradation des résultats au cours du processus. Les seuil $0,9$ et $0,99$ produisent les meilleurs résultats. Sur le sous-ensemble des pages collectées, les résultats sont nettement moindres, bien que toujours largement au-dessus d'une classification aléatoire ou naïve [11]. Cette difficulté accrue sur les données collectées automatiquement provient probablement d'éléments que le nettoyage automatique

10. Souvent à cause d'un mélange de registres entre des propos rapportés et des passage de narration.

11. Taux de bonne classification de 36 % dans le cas d'un tirage aléatoire informé sur la distribution des classe, 50 % dans le cas du vote majoritaire (classe « courant »).

	(a) Test			(b) Pages collectées annotées		
	Familier	Courant	Soutenu	Familier	Courant	Soutenu
Rappel	$0,90$	$0,78$	$0,93$	$0,53$	$0,72$	$0,45$
Précision	$0,84$	$0,90$	$0,87$	$0,52$	$0,64$	$0,61$
F-mesure	$0,87$	$0,83$	$0,90$	$0,52$	$0,68$	$0,52$

TABLE 3 – Rappel, précision et F-mesure pour chaque registre en fin de processus (seuil = $0,99$) sur l'ensemble de test (a) et sur le sous-ensemble manuellement annoté des pages collectées (b).

n'a pas réussi à enlever [12]. Les conclusions générale sur l'impact des différents seuils sont identiques. En complément, il est intéressant de noter que le seuil de $0,99$ produit une augmentation du taux de bonne classification lors des 2 premières étapes de sélection, puis baisse progressivement. Ce comportement pose la question de la détermination automatique d'un critère d'arrêt des itérations. En l'état, des analyses plus approfondies sont nécessaires pour mieux comprendre les phénomènes observés et élaborer un critère de qualité globale du corpus étiqueté après chaque étape.

La table 3 présente les taux de rappel et précision ainsi que la F-mesure en fin de processus pour le seuil $0,99$, sur l'ensemble de test et sur notre extrait annoté des pages collectées. Sur le premier ensemble, il apparaît que ces mesures sont relativement homogènes entre registres. La F-mesure la plus basse est celle du registre courant. Elle s'explique notamment par un rappel plus bas que pour les autres registres. Les résultats sont à l'inverse sur l'extrait annoté des pages web puisque ils sont globalement beaucoup plus faibles (conformément aux résultats de la figure 3) et que le registre courant est celui le mieux reconnu. Les registres familier et soutenu présentent eux une F-mesure très faible mais pour des raisons différentes. Pour le premier, le modèle semble retourner avoir des difficultés globales avec un rappel une précision à peine supérieurs à la moyenne alors que, pour le second, les faibles résultats semblent davantage liés à un forte proportion de faux négatifs (rappel faible). Dans l'optique d'une utilisation fiable du corpus construit, il apparaît donc nécessaire d'améliorer encore ces résultats. Pour cela, une attention particulière devra notamment être porté sur la limitation des faux positifs car ceux-ci tendent probablement à freiner ou fausser la convergence du processus semi-supervisé.

6.2 Corpus étiqueté automatiquement

Les figures 4 et 5 illustrent la construction du corpus étiqueté. La première présente l'évolution de la taille de ce corpus pour les différents seuils étudiés, la seconde la proportion de chaque registre dans celui-ci dans l'unique meilleur cas d'un seuil de sélection fixé à $0,99$. En terme de taille, il apparaît, d'une part, que la totalité ou quasi totalité des pages collectées termine le processus avec une étiquette. Étant donné le bruit déjà évoqué dans les données, ce constat semble à nouveau indiquer la nécessité d'un critère d'arrêt du processus. D'autre part, il apparaît que l'étiquetage intégral des données se produit rapidement, c'est-à-dire en peu d'itérations. Par exemple, 89% des étiquettes sont validées à l'issue de la première passe pour un seuil de sélection de $0,8$. Ceci témoigne de la grande confiance du modèle dans ses prédictions. Nous pensons que ceci peut s'expliquer par une importance trop grande donnée à certaines descripteurs (par exemple, l'apparition ou l'absence de termes d'un lexique spécifique) pour prédire un registre. Une solution pourrait être d'introduire un mécanisme d'abandon

12. Par exemple, dans le cas de forums où de nombreux éléments textuels sont à supprimer pour n'isoler que le corps des réponses.

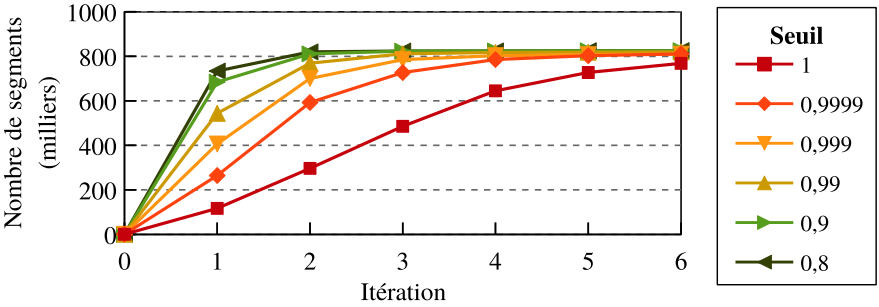


FIGURE 4 – Taille du corpus pour chaque itération.

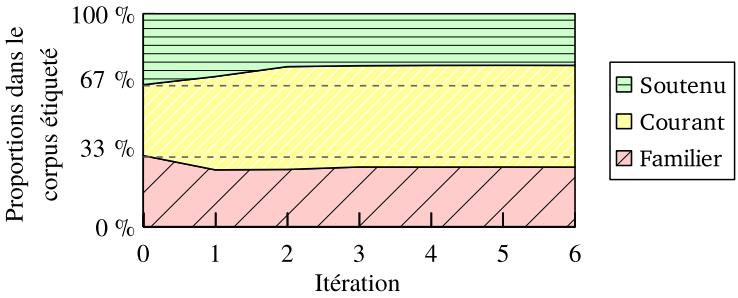


FIGURE 5 – Évolution de la proportion de chaque registre dans le corpus étiqueté (pourcentage sur le nombre de textes, seuil = $0,99$).

(*dropout*) lors de l'apprentissage du réseau de neurones pour l'amener à des prédictions s'appuyant sur un spectre plus large d'informations. De premières expériences ont été conduites dans cette direction mais il apparaît que cette stratégie peut conduire à des dégradation du taux de classification lorque le taux d'abandon des informations est mal configuré. Enfin, l'évolution de la répartition des classes est également intéressante à observer puisque nous avons montré à travers l'annotation de quelques pages collectées que la répartition des registres diffère entre notre graine, volontairement équilibrée, et les pages collectée, avec un forte dominance du registre courant. La figure 5 montre que l'étiquetage semi-supervisée de notre approche corrige de lui-même cette différence. Dans le corpus final, le registre courant représente $48\,\%$ des étiquettes, contre respectivement $28\,\%$ et $24\,\%$ pour les registres familier et soutenu. Ces nombres coïncident globalement avec ceux de notre étiquetage manuel sur un petit échantillon aléatoire.

En fin de processus pour le seuil de $0,99$, les textes annotés comme familiers viennent pour $68\,\%$ des requêtes construites sur le lexique familier et, donc, pour $32\,\%$ de celles sur le lexique soutenu. Ces rapports sont respectivement de $47\,\%$ / $53\,\%$ et $37\,\%$ / $63\,\%$ pour le corpus des registres courant et soutenu. La table 4 montre ainsi 2 extraits de pages issues de requêtes soutenues mais l'une ayant été étiquetée comme du registre courant et l'autre soutenu. D'une part, ces proportions montrent que les requêtes formées à partir des lexiques facilitent la construction du corpus par un amorçage approprié du processus puisqu'elles n'enferment pas les pages récoltées dans le registre de leur requête d'origine. Une conclusion intéressante est donc que la présence de termes discriminants pour un registre n'est pas un indice suffisant pour catégoriser un texte dans ledit registre. En cela, cette

Courant	Soutenu
Oui, Monsieur Adrien Richard, si vous aimez mieux, le directeur de l'usine, mais nous, nous ne l'appelons que Monsieur Adrien, parce qu'on a été à l'école ensemble et qu'il nous appelle aussi par notre prénom.	D'ailleurs, nous retrouvons la même distinction dédaigneuse à l'égard des professionnels et de leur " vil salaire " qui ne les empêche pas de mourir " ès hôpitaux ", chez le docte Muret.

TABLE 4 – Rappel, précision et F-mesure pour chaque registre en fin de processus (seuil = $0,99$) sur l'ensemble de test (a) et sur le sous-ensemble manuellement annoté des pages collectées (b).

souplesse justifie également le recours à une classification des pages récoltées. D'autre part, l'analyse montre que certains phénomènes sont encore mal compris et que la méthode devrait être affinée. En particulier, il apparaît que la plupart des textes classés comme familiers à partir de requêtes soutenues (et réciproquement) ne devrait pas l'être. Hormis une règle stricte qui interdirait ces situations, il semble nécessaire d'observer les traits qui contribuent à ces erreurs et d'affiner la méthode actuel, par exemple en considérant des descripteurs complémentaires ou plus précis. Par exemple, il apparaît que les mots vulgaires, confondus dans l'ensemble des termes familiers, ont un rôle ambigü. De même, les fréquences de ponctuation ou la longueur des phrases devraient être prises en compte.

7 Conclusion

Dans cet article, nous avons présenté un processus semi-supervisé qui construit conjointement un corpus textuel étiqueté en registres de langue et un classifieur associé. En s'appuyant sur un très large ensemble de textes et quelques ressources expertes de départ, le résultat de cette approche est un corpus constitué de $800\,000$ segments textuels représentant un total d'environ 750 millions de mots. Le classifieur parallèlement obtenu atteint un taux de bonne classification de $87\,\%$ sur l'ensemble de test mais des résultats plus modestes sur un sous-ensemble étiqueté manuellement des pages collectées. Ces résultats semblent démontrer la validité de l'approche et d'une majorité des annotations produites mais ils démontrent également le besoin d'affiner ses différents aspects.

Parmi les pistes de travail pour l'avenir, une analyse linguistique poussée sur la qualité des annotations automatiques doit être poursuivie, ainsi qu'une étude des importances de chaque descripteur dans le réseau de neurones et de la robustesse du réseau si certains descripteurs venaient à être absents ou anormaux. Par ailleurs, le choix du seuil de sélection ne semble pas critique ni même réellement propice à éviter la présence trop importante de faux positifs. Il serait instructif de comprendre pourquoi et d'essayer d'autres stratégies (par exemple, en limitant la sélection des textes à un nombre fixe par itération). À plus long terme, le corpus ouvre de multiples pistes d'utilisation, comme, dans le cas qui nous intéressent, la transposition automatique d'un texte d'un registre vers un autre.

Remerciements

Ce travail a bénéficié du soutien financier de l'Agence Nationale de la Recherche (ANR) dans le cadre du projet TREMoLo (ANR-16-CE23-0019).

Références

ARGAMON S., WHITELAW C., CHASE P., HOTA S. R., GARG N. & LEVITAN S. (2007). Stylistic text classification using functional lexical features. *Journal of the Association for Information Science and Technology*, **58**(6), 802–822.

BARONI M. & BERNARDINI S. (2004). Bootcat : Bootstrapping corpora and terms from the web. In *Proceedings of the Language Resources and Evaluation Conference (LREC)*, p. 1313–1316.

BIBER D. & CONRAD S. (2009). *Register, genre, and style*. Cambridge University Press.

BIBER D. & FINEGAN E. (1994). *Sociolinguistic perspectives on register*. Oxford University Press on Demand.

BORZEIX A. & FRAENKEL B. (2005). *Langage et travail (communication, cognition, action)*. CNRS éd.

CHARAUDEAU P. (1997). *Le discours d'information médiatique : la construction du miroir social*. Nathan.

COUGNON L.-A. & FAIRON C. (2014). *SMS Communication : A linguistic approach*, volume 61. John Benjamins Publishing Company.

DE VEL O., ANDERSON A., CORNEY M. & MOHAY G. (2001). Mining e-mail content for author identification forensics. *ACM Sigmod Record*, **30**(4), 55–64.

EISENSTEIN J. (2013). What to do about bad language on the internet. In *Proceedings of North American Chapter of the Association for Computational Linguistics : Human Language Technologies (HLT-NAACL)*, p. 359–369.

ESCALANTE H. J., SOLORIO T. & MONTES-Y GÓMEZ M. (2011). Local histograms of character n-grams for authorship attribution. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics : Human Language Technologies (HTL-ACL)*, p. 288–298 : Association for Computational Linguistics.

GADET F. (1996). Niveaux de langue et variation intrinsèque. *Palimpsestes*, **10**, 17–40.

GIANFORTONI P., ADAMSON D. & ROSÉ C. P. (2011). Modeling of stylistic variation in social media with stretchy patterns. In *Proceedings of the Workshop on Algorithms and Resources for Modelling of Dialects and Language Varieties*, p. 49–59 : Association for Computational Linguistics.

HIRST G. & FEIGUINA O. (2007). Bigrams of syntactic labels for authorship discrimination of short texts. *Literary and Linguistic Computing*, **22**(4), 405–417.

IQBAL F., BINSALLEEH H., FUNG B. C. & DEBBABI M. (2013). A unified data mining solution for authorship analysis in anonymous textual communications. *Information Sciences*, **231**, 98–112.

KOBUS C., YVON F. & DAMNATI G. (2008). Normalizing sms : are two metaphors better than one ? In *Proceedings of the International Conference on Computational Linguistics (COLING)*, p. 441–448 : Association for Computational Linguistics.

KOPPEL M. & SCHLER J. (2003). Exploiting stylistic idiosyncrasies for authorship attribution. In *Proceedings of IJCAI Workshop on Computational Approaches to Style Analysis and Synthesis*, volume 69, p. 72–80.

LE Q. & MIKOLOV T. (2014). Distributed representations of sentences and documents. In *Proceedings of the International Conference on Machine Learning (ICML)*, p. 1188–1196.

LECORVÉ G., GRAVIER G. & SÉBILLOT P. (2008). On the use of web resources and natural language processing techniques to improve automatic speech recognition systems. In *Proceedings of the Language Resources and Evaluation Conference (LREC)*, p. 592–599.

MARTON Y., WU N. & HELLERSTEIN L. (2005). On compression-based text classification. In *Proceedings of the European Conference on Information Retrieval (ECIR)*, volume 3408, p. 300–314 : Springer.

MCCARTHY P. M., LEWIS G. A., DUFTY D. F. & MCNAMARA D. S. (2006). Analyzing writing styles with coh-metrix. In *Proceedings of the FLAIRS Conference*, p. 764–769.

MEKKI J., BATTISTELLI D., BÉCHET N. & LECORVÉ G. (2017). *« Nous nous arrachâmes promptement avec ma caisse » : quels descripteurs linguistiques caractérisent les registres de langue ?* Technical report, IRISA, équipe EXPRESSION ; MoDyCo.

MEKKI J., BATTISTELLI D., LECORVÉ G. & BÉCHET N. (2018). Identification de descripteurs pour la caractérisation de registres. In *Actes des Rencontres Jeunes Chercheurs (RJC) de la conférence CORIA-TALN*.

MIKOLOV T., YIH W.-T. & ZWEIG G. (2013). Linguistic regularities in continuous space word representations. In *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics : Human Language Technologies (HLT-NAACL)*, p. 746–751.

MOIRAND S. (2007). *Les discours de la presse quotidienne. Observer, analyser, comprendre*. Puf.

SANDERS C. (1993). *Sociosituational variation*. Cambridge : Cambridge University Press.

SANDERSON C. & GUENTER S. (2006). Short text authorship attribution via sequence kernels, markov chains and author unmasking : An investigation. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, p. 482–491 : Association for Computational Linguistics.

SCHLER J., KOPPEL M., ARGAMON S. & PENNEBAKER J. W. (2006). Effects of age and gender on blogging. In *Proceedings of the AAAI spring symposium : Computational approaches to analyzing weblogs*, volume 6, p. 199–205.

SIDOROV G., VELASQUEZ F., STAMATATOS E., GELBUKH A. & CHANONA-HERNÁNDEZ L. (2014). Syntactic n-grams as machine learning features for natural language processing. *Expert Systems with Applications*, **41**(3), 853–860.

STAMATATOS E. (2009). A survey of modern authorship attribution methods. *Journal of the Association for Information Science and Technology*, **60**(3), 538–556.

TAMBOURATZIS G., MARKANTONATOU S., HAIRETAKIS N., VASSILIOU M., CARAYANNIS G. & TAMBOURATZIS D. (2004). Discriminating the registers and styles in the modern greek language-part 2 : Extending the feature vector to optimize author discrimination. *Literary and Linguistic Computing*, **19**(2), 221–242.

URE J. (1982). Introduction : approaches to the study of register range. *International Journal of the Sociology of Language*, **1982**(35), 5–24.

Approche supervisée à base de cellules *LSTM* bidirectionnelles pour la désambiguïsation lexicale

Loïc Vial Benjamin Lecouteux Didier Schwab

Univ. Grenoble Alpes, CNRS, Grenoble INP, LIG, 38000 Grenoble, France
{loic.vial, benjamin.lecouteux, didier.schwab}@univ-grenoble-alpes.fr

RÉSUMÉ

En désambiguïsation lexicale, l'utilisation des réseaux de neurones est encore peu présente et très récente. Cette direction est pourtant très prometteuse, tant les résultats obtenus par ces premiers systèmes arrivent systématiquement en tête des campagnes d'évaluation, malgré une marge d'amélioration qui semble encore importante. Nous présentons dans cet article une nouvelle architecture à base de réseaux de neurones pour la désambiguïsation lexicale. Notre système est à la fois moins complexe à entraîner que les systèmes neuronaux existants et il obtient des résultats état de l'art sur la plupart des tâches d'évaluation de la désambiguïsation lexicale en anglais. L'accent est porté sur la reproductibilité de notre système et de nos résultats, par l'utilisation d'un modèle de vecteurs de mots, de corpus d'apprentissage et d'évaluation librement accessibles.

ABSTRACT

LSTM Based Supervised Approach for Word Sense Disambiguation

In word sense disambiguation, there are still few usages of neural networks. This direction is very promiseful however, the results obtained by these first systems being systematically in the top of the evaluation campaigns, with an improvement gap which seems still high. We present in this paper a new architecture based on neural networks for word sense disambiguation. Our system is at the same time less difficult to train than existing neural networks, and it obtains state of the art results on most evaluation tasks in English. The focus is on the reproducibility of our systems and our results, through the use of a word embeddings model, training corpora and evaluation corpora freely accessible.

MOTS-CLÉS : Désambiguïsation lexicale, Approche supervisée, LSTM, Réseau neuronal.

KEYWORDS: Word Sense Disambiguation, Supervised Approach, LSTM, Neural Network.

1 Introduction

La Désambiguïsation Lexicale (DL) est une tâche centrale en Traitement Automatique des Langues (TAL) qui vise à attribuer le sens le plus probable à un mot donné dans un document, à partir d'un inventaire prédéfini de sens.

Il existe une multitude d'approches pour la DL, dont les approches supervisées, qui utilisent des méthodes d'apprentissage automatique couplées à de grandes quantités de données manuellement annotées, les approches à base de connaissances, qui se basent sur des ressources lexicales telles que

des dictionnaires, des thésaurus ou des réseaux lexicaux par exemple, les approches semi-supervisées, non-supervisées, ou encore les approches à base de graphes ou de similarités. Pour un état de l'art plus complet, le lecteur est invité à lire par exemple Navigli (2009).

Depuis la création des campagnes d'évaluation pour les systèmes de DL telles que SensEval/SemEval, les approches supervisées se retrouvent systématiquement dans les premières places en terme de scores obtenus (Chan *et al.*, 2007; Zhong & Ng, 2010; Iacobacci *et al.*, 2016). Alors que l'on voit se multiplier les utilisations de techniques d'apprentissage à base de réseaux de neurones dans la plupart des champs de recherche du TAL, comme par exemple pour la représentation vectorielle des mots (Mikolov *et al.*, 2013; Pennington *et al.*, 2014; Bojanowski *et al.*, 2017), la traduction automatique (Sutskever *et al.*, 2014; Cho *et al.*, 2014) ou l'étiquetage morpho-syntaxique (Andor *et al.*, 2016), on retrouve aussi des approches supervisées à base de réseaux de neurones pour la désambiguïsation lexicale, et ce sont ces méthodes qui obtiennent aujourd'hui les résultats état de l'art (Yuan *et al.*, 2016; Kågebäck & Salomonsson, 2016; Raganato *et al.*, 2017b).

Dans cet article, nous présentons une nouvelle approche supervisée de DL à base de réseaux de neurones, qui s'appuie sur les modèles existants et qui obtient des résultats état de l'art sur la plupart des tâches d'évaluation de la DL en anglais tout en étant moins complexe et difficile à mettre en place. De plus, nous utilisons pour la première fois l'ensemble des corpus annotés avec des sens provenant de la base lexicale *WordNet* (Miller, 1995) qui existent à ce jour, ce qui permet à notre système d'être plus robuste car plus généralisable à de nouvelles données.

En effet, les systèmes supervisés de l'état de l'art sont généralement uniquement entraînés sur le SemCor (Miller *et al.*, 1993), mais une demi-douzaine d'autres corpus annotés en sens et de grande taille existent. Notre équipe les a tous regroupés dans une ressource libre nommée UFSAC [1] (Vial *et al.*, 2017). Par soucis de comparaison avec les systèmes état de l'art, nous avons évalué notre approche à la fois en utilisant tous les corpus UFSAC disponibles, mais aussi en nous restreignant uniquement au SemCor.

Dans un premier temps nous allons présenter les architectures des systèmes neuronaux de DL de l'état de l'art, avec leurs avantages et inconvénients respectifs dans la section 2, ainsi que l'architecture que nous proposons dans la section 3. Ensuite nous décrirons le protocole expérimental que nous avons suivi pour évaluer notre système dans la section 4 puis nous détaillerons les résultats dans la section 5. Enfin nous présenterons un travail préliminaire d'amélioration de notre système de manière totalement non supervisée dans la section 6 et enfin nous conclurons dans la section 7.

2 Architectures neuronales pour la désambiguïsation lexicale

Parmi les approches neuronales pour la DL, on retrouve notamment trois travaux majeurs : le modèle de Kågebäck & Salomonsson (2016), le modèle de Yuan *et al.* (2016) et celui de Raganato *et al.* (2017b).

Kågebäck & Salomonsson (2016) sont les premiers à mettre en œuvre un réseau de neurones à base de vecteurs de mots et de cellules récurrentes de type *LSTM* pour prédire le sens d'un mot cible. Dans leurs travaux, un modèle n'est capable de prédire le sens que d'un seul lemme du dictionnaire, et donc chaque lemme a son modèle propre de classification qui est entraîné séparément. Leur système est évalué sur les tâches de *lexical sample* des campagnes d'évaluation SensEval 2 et SensEval 3 dans

1. https://github.com/getalp/UFSAC

lesquelles plusieurs instances d'un faible nombre de lemmes distincts sont à annoter en sens, mais il n'est pas évalué sur les tâches de désambiguïsation lexicale *all words* où tous les mots d'un document doivent être annotés en sens.

Le principal avantage de leur modèle est donc sa petite taille. En effet la couche de sortie de leur réseau est de la taille du nombre de sens pour le lemme cible, le nombre de sens moyen pour les mots polysémiques dans WordNet étant d'environ 3 [2]. Les couches cachées de cellules *LSTM* sont elles aussi très petites, avec seulement deux couches de taille 74 chacune. Il est cependant peu aisé d'entraîner ce système à annoter tous les mots d'un document car chaque lemme doit avoir son propre modèle.

Dans le modèle de Yuan *et al.* (2016), un réseau neuronal à base de cellules *LSTM* est utilisé comme modèle de langue, pour prédire un mot d'une séquence en fonction de son contexte. Un apprentissage supervisé sur des corpus annotés en sens est ensuite effectué pour que leur système apprenne à distinguer les différents sens d'un mot en fonction des mots prédits par leur modèle de langue. Dans un second temps, les auteurs proposent une méthode de propagation de labels pour augmenter leurs données annotées en sens et obtenir ainsi leurs meilleurs résultats. Cette méthode consiste à chercher dans des corpus non annotés de nouvelles phrases, proches des phrases de leur corpus annoté, en se basant sur une mesure de similarité cosinus entre les représentations vectorielles de ces phrases. Les annotations en sens sont ensuite propagées de la phrase initialement annotée vers l'autre phrase.

Dans cet article, les auteurs comparent les performances de différents modèles, entraînés sur le SemCor ou l'OMSTI, avec et sans leur propagation de labels, et obtiennent des résultats état de l'art sur la plupart des tâches. Le principal problème de leur approche est la reproductibilité des résultats, en effet leur modèle de langue est entraîné sur un corpus privé d'actualités (*news*) d'une taille de 100 milliards de mots, et ils ont utilisé pour leur propagation de labels des phrases prises aléatoirement sur le Web, sans en spécifier la source plus précisément.

Enfin, l'architecture de leur modèle de langue ne permet de prédire le sens que d'un seul mot à la fois pour une séquence donnée, parce que le mot cible doit être remplacé par un symbole spécial avant d'être donné en entrée de leur réseau. Il est donc nécessaire d'exécuter leur modèle pour chaque mot d'une phrase afin de tous les annoter.

Raganato *et al.* (2017b) proposent également un modèle à base de *LSTM* mais qui apprend directement à prédire un label pour chacun des mots donnés en entrée. Le label à prédire fait partie d'un ensemble comprenant tous les sens possibles dans un dictionnaire ainsi que tous les mots observés pendant l'entraînement. Ils augmentent ensuite leur modèle avec une couche d'attention, et ils effectuent un entraînement multi-tâches dans lequel leur réseau prédit à la fois un sens ou un mot, un label de partie du discours, et un label sémantique.

Cette architecture est la seule qui permet d'annoter tous les mots d'une séquence en une passe et l'entraînement de leur modèle s'est effectué sur le SemCor uniquement. Leur réseau associe à un mot en entrée un label appartenant à l'ensemble des sens de leur inventaire de sens ainsi que l'ensemble des mots observés pendant l'entraînement. Cette approche permet à leur modèle d'apprendre à prédire un label de sens lorsque le mot est annoté dans le corpus d'entraînement, et un label de mot lorsque le mot n'est pas annoté (si c'est un mot outil par exemple). L'inconvénient de leur approche est qu'elle n'est pas applicable lorsque l'on veut réaliser l'apprentissage sur un corpus partiellement annoté en sens. En effet pour ce type de corpus, leur modèle va apprendre à "recopier" des mots non annotés alors qu'ils sont potentiellement porteurs de sens.

2. https://wordnet.princeton.edu/documentation/wnstats7wn

3 Architecture proposée

Notre approche est, comme pour Raganato *et al.* (2017b), de considérer la désambiguïsation lexicale comme un problème de classification dans lequel un label est assigné à chaque mot. Cependant, nous simplifions leur modèle en considérant un label comme appartenant uniquement à l'ensemble de tous les sens possibles de notre inventaire de sens. L'architecture de notre réseau de neurones, illustrée par la figure 1 repose ainsi sur 3 couches de cellules :

— La couche d'entrée, qui prend directement les mots sous une forme vectorielle construite séparément de notre système. On pourra utiliser ici n'importe quelle base de vecteurs de mots pré-entraînés telle que Word2Vec (Mikolov *et al.*, 2013) ou GloVe (Pennington *et al.*, 2014).

— La couche cachée, composée de cellules LSTM (Hochreiter & Schmidhuber, 1997) bidirectionnelles. Ces cellules dites "à mémoire" aussi appelées cellules "récurrentes" permettent de calculer une sortie en considérant non seulement l'élément courant de la séquence, mais aussi l'historique passé des cellules précédentes. Ces cellules sont communément utilisées pour l'apprentissage automatique sur des séquences, que ce soit sur du texte écrit (Sutskever *et al.*, 2014) ou de la parole (Chan *et al.*, 2016).

— La couche de sortie, qui génère pour chacun des mots en entrée, une distribution de probabilité sur tous les sens possibles du dictionnaire, à l'aide d'une fonction softmax classique.

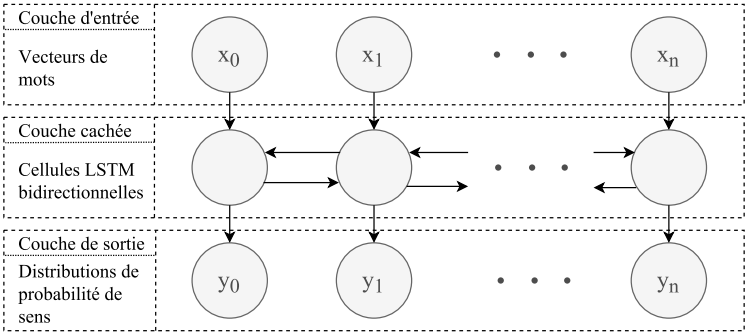


FIGURE 1 – Architecture de notre réseau de neurones pour la DL.

La fonction de coût à minimiser pendant la phase d'apprentissage est l'entropie croisée entre la couche de sortie et un vecteur de type *one-hot*, pour lequel toutes les composantes sont à 0 sauf à l'index du sens cible où elle est à 1. On cherche ainsi à minimiser la fonction $H(p,q) = -\sum_x p(x) \log q(x)$, où x est est une composante du vecteur de la couche de sortie, p est la distribution de probabilité réelle et q la sortie de notre réseau de neurones. Comme toutes les valeurs de la distribution réelle sont à 0 sauf à l'index du sens correct, pour un exemple donné, on cherche ainsi à minimiser la formule $-\log q(s)$, où s est l'index du sens à prédire.

Notre modèle prédit toujours un sens en sortie pour chaque mot en entrée, même pour les mots outils ou les mots qui n'ont pas été annotés dans le corpus d'entraînement, cependant, dans ces cas là, nous avons un symbole spécial <*skip*> nous permettant d'ignorer les prédictions faites par le modèle et de ne pas en tenir compte lors de la phase de rétro-propagation durant l'entraînement.

Contrairement à l'approche proposée par Raganato *et al.* (2017b), notre modèle peut ainsi apprendre non seulement sur des données entièrement annotées, comme c'est le cas avec le SemCor (Miller

et al., 1993) par exemple, mais également sur des données partiellement annotées, comme l'OMSTI (Taghipour & Ng, 2015) ou le DSO (Ng & Lee, 1997), dans lesquelles un seul mot est annoté par phrase. Il est en effet capable d'apprendre à prédire les sens de tous les mots d'une séquence en même temps, et à la fois d'en ignorer certains éléments. L'entraînement se retrouve aussi moins complexe à réaliser que pour Raganato *et al.* (2017b) car la taille de la couche de sortie est beaucoup plus petite : le nombre de sens différents dans la version 3.0 de *WordNet* est de 117 659 [3], alors qu'une taille de vocabulaire typique pour des modèles de vecteurs de mots en anglais contient au minimum 400 000 mots et plus généralement plus de 1 000 000 de mots [4] [5].

Notre architecture est aussi très différente de celles de Yuan *et al.* (2016) ou de Kågebäck & Salomonsson (2016), notamment car leurs architectures ne permettent pas d'annoter tous les mots en entrée de leurs modèles en une seule passe, mais seulement indépendamment les uns des autres.

4 Protocole expérimental

Pour évaluer notre système de DL à base de réseaux de neurones, nous avons tiré parti de notre précédent travail (Vial *et al.*, 2017) dans lequel nous proposions une ressource contenant tous les corpus anglais annotés en sens *WordNet* connus à ce jour, et nous avons entraîné notre modèle sur 6 de ces corpus : le SemCor (Miller *et al.*, 1993), le DSO (Ng & Lee, 1997), le WordNet Gloss Tagged (Miller, 1995), l'OMSTI (Taghipour & Ng, 2015), le MASC (Ide *et al.*, 2008) et l'Ontonotes (Hovy *et al.*, 2006). Nous avons utilisé le corpus de la tâche 13 de SemEval 2015 (Moro & Navigli, 2015) comme corpus de développement durant l'apprentissage, pour éviter le surapprentissage de nos données d'entraînement. Enfin, nous avons évalué le modèle ayant obtenu le meilleur score F1 de DL sur notre corpus de développement, sur les corpus de SensEval 2 (Edmonds & Cotton, 2001), SensEval 3 (Snyder & Palmer, 2004), les tâches 7 et 17 de SemEval 2007 (Navigli *et al.*, 2007; Pradhan *et al.*, 2007), et enfin la tâche 12 de SemEval 2013 (Navigli *et al.*, 2013).

Pour comparer l'architecture que nous proposons avec l'état de l'art, et notamment Raganato *et al.* (2017b) et Yuan *et al.* (2016) qui utilisent uniquement le SemCor comme corpus d'apprentissage supervisé, nous avons aussi évalué notre approche en limitant l'apprentissage du modèle à ce corpus.

Dans certains corpus, les mots peuvent être annotés avec plusieurs sens *WordNet*, soit parce que l'annotateur a trouvé qu'ils étaient tous applicables, ou bien parce que les sens ont été initialement annotés avec un autre dictionnaire puis convertis en sens *WordNet* (c'est le cas du MASC par exemple). Dans ce cas nous supprimons toutes les annotations pour ne garder au final que les annotations qui ne contiennent qu'un seul sens dans notre corpus d'apprentissage.

En entrée de notre réseau, nous avons utilisé les vecteurs de GloVe (Pennington *et al.*, 2014) pré-entrainés sur Wikipedia 2014 et Gigaword 5 disponibles librement [6]. La taille des vecteurs est de 300, la taille du vocabulaire est de 400 000 et tous les mots sont mis en minuscules. Nous avons choisi ces vecteurs pour la petite taille de leur modèle pré-entraîné et pour sa qualité par rapport aux tâches de similarité de mots et d'analogie de mots. Ce sont aussi ces vecteurs qui sont utilisés en entrée du réseau décrit par Kågebäck & Salomonsson (2016).

3. https://wordnet.princeton.edu/documentation/wnstats7wn
4. https://nlp.stanford.edu/projects/glove/
5. https://fasttext.cc/docs/en/english-vectors.html
6. https://nlp.stanford.edu/projects/glove/

Pour la couche cachée de neurones récurrents, nous avons choisi des cellules *LSTM* de taille de 1000 par direction (donc 2000 au total). C'est à peu près la taille qui est utilisée dans Raganato *et al.* (2017b) (chaque *LSTM* est de taille 1024) et Yuan *et al.* (2016) (une seule couche de taille 2048).

Enfin, entre la couche cachée et la couche de sortie, nous avons appliqué une régularisation de type *Dropout* (Srivastava *et al.*, 2014) à 50%, une méthode classique qui vise à empêcher le surapprentissage pendant l'entraînement afin de rendre le modèle plus robuste.

Cette configuration permet de reproduire aisément nos résultats. En effet, en plus du modèle de vecteurs de mots pré-entraîné, tous les corpus utilisés sont libres d'accès et dans un format unifié [7]. La seule exception est le corpus DSO qui est payant, il ne contient cependant qu'approximativement 8% des mots annotés dans nos corpus d'apprentissage, avec seulement 121 noms et 70 verbes différents.

Les paramètres utilisés pour l'apprentissage sont les suivants :

— La méthode d'optimisation est Adam (Kingma & Ba, 2014), avec les mêmes paramètres par défaut tels que décrits dans leur article, c'est à dire $\alpha = 0.001$, $\beta_1 = 0.9$, $\beta_2 = 0.999$ et $\epsilon = 10^{-8}$;

— la taille de mini-lots utilisée est de 30 ;

— les phrases sont tronquées à 50 mots, pour faciliter l'entraînement tout en minimisant la perte d'informations (moins de 5% des mots annotés dans nos données d'entraînement sont perdus) ;

— les séquences sont remplies de vecteurs nuls depuis la fin de façon à ce qu'elles aient toutes la même taille au sein d'un mini-lot.

Nous avons construit notre réseau neuronal à l'aide de l'outil *PyTorch* [8] et nous avons effectué l'apprentissage pendant 20 *epochs*. Une *epoch* correspondant à une passe complète sur nos données d'entraînement. Nous avons évalué périodiquement (tous les 2000 mini-lots et à la fin de chaque *epoch*) notre modèle sur le corpus de développement, et nous avons conservé uniquement le modèle ayant obtenu le plus grand score F1 de désambiguïsation.

Pour réaliser la désambiguïsation d'une séquence de mots en utilisant le réseau entraîné, la méthode suivante est utilisée :

1. Chaque mot est d'abord transformé en vecteur à l'aide du modèle de vecteurs de mots, puis donné en entrée au réseau.
2. En sortie, une distribution de probabilité sur tous les sens observés pendant l'apprentissage est retournée pour chaque élément de la séquence. Nous assignons le sens le plus probable en suivant cette distribution, parmi les sens possibles du mot dans *WordNet*, en fonction de son lemme et de sa partie du discours. Ces deux informations étant systématiquement données pendant les campagnes d'évaluation de la DL.
3. Si aucun sens n'est assigné, une stratégie de repli est effectuée. La plus courante et celle que nous utilisons ici est d'assigner au mot son sens le plus fréquent dans *WordNet*.

Le processus d'apprentissage est forcément stochastique, en effet non seulement les poids du modèle sont initialisés aléatoirement par la bibliothèque sous-jacente, mais le corpus d'apprentissage est également mélangé à chaque début d'*epoch*. Nous avons entraîné ainsi 8 modèles séparément pour chacun de nos tests, puis nous avons utilisé une moyenne géométrique sur toutes les prédictions faites par ces modèles pour obtenir la distribution de sens finale que nous avons utilisée pour réaliser une désambiguïsation. C'est une pratique couramment utilisée (par exemple (Sutskever *et al.*, 2014)) car elle permet non seulement d'avoir un système moins sensible au bruit et donc plus robuste, mais

7. https://github.com/getalp/UFSAC
8. http://pytorch.org/

aussi un système de meilleure qualité. En effet, un modèle peut être individuellement bloqué dans un minimum local pendant l'entraînement et avoir un très bon score sur le corpus de développement, mais être incapable de généraliser, alors qu'il est improbable que ce problème arrive à l'ensemble de modèles.

5 Résultats

Nous avons évalué notre modèle sur tous les corpus d'évaluation communément utilisés en DL, à savoir les tâches de DL des campagnes d'évaluation SensEval/SemEval. Les scores obtenus par notre système comparés à ceux des systèmes semblables de l'état de l'art à base de réseaux de neurones (Yuan *et al.*, 2016; Raganato *et al.*, 2017b), ainsi que l'étalon du sens le plus fréquent, et du meilleur système précédant l'utilisation des réseaux de neurones en DL (Iacobacci *et al.*, 2016) se trouvent dans la table 1.

Système	SE2	SE3	SE07 (07)	SE07 (17)	SE13 (12)	SE15 (13)
Notre système (SemCor)	68.27	69.95	80.14	60.51	62.97	*69.72
Notre système (SemCor + repli)	73.71	**71.68**	**83.99**	**61.98**	67.58	*72.74
Notre système (UFSAC)	72.54	69.46	82.87	59.85	67.53	*73.56
Notre système (UFSAC + repli)	**73.75**	70.16	83.59	60.00	**68.92**	***73.98**
Yuan *et al.* (2016) (LSTM)	73.6	69.2	82.8	**64.2**	67.0	72.1
Yuan *et al.* (2016) (LSTM + LP)	**73.8**	**71.8**	**83.6**	63.5	**69.5**	**72.6**
Raganato *et al.* (2017b) (BLSTM)	71.4	68.8	-	*61.8	65.6	69.2
Raganato *et al.* (2017b) (BLSTM + att. + LEX + POS)	72.0	69.1	83.1	*64.8	66.9	71.5
Sens le plus fréquent	65.6	66.0	78.89	54.5	63.8	67.1
Iacobacci *et al.* (2016)	68.3	68.2	-	59.1	-	-

TABLE 1 – Scores F1 (%) obtenus par notre système sur les tâches de DL des campagnes d'évaluation SensEval 2 (SE2), SensEval 3 (SE3), SemEval 2007 (SE07) tâches 07 et 17, SemEval 2013 (SE13) tâche 12 et SemEval 2015 (SE15) tâche 13. Les résultats préfixés par un astérisque (*) sont obtenus sur le corpus utilisé pour le développement pendant l'apprentissage. Les résultats affichés en gras sont les meilleurs obtenus par notre système et par les systèmes de l'état de l'art. Les résultats affichés en rouge sont les meilleurs de l'état de l'art.

Pour toutes les tâches, nous avons évalué notre système avec et sans le repli sur le premier sens pour les mots qui n'ont pas été observés pendant l'apprentissage. Nous l'avons aussi évalué dans un premier temps avec un apprentissage sur le SemCor uniquement, et sur les 6 corpus UFSAC combinés dans un second temps.

On remarque d'abord qu'en termes de scores F1 avec le repli, il y a très peu de différences entre le système entraîné sur le SemCor et celui entraîné sur tous les corpus UFSAC. Nos meilleurs résultats sur les tâches de SensEval 3 et de SemEval 2007 sont même obtenus par le système qui est entraîné sur le SemCor uniquement. Le SemCor possède pourtant seulement environ 10% des mots annotés dans UFSAC.

Cependant, lorsque l'on compare les scores de désambiguïsation d'un système avec repli et sans

repli, la différence entre ces deux scores est bien plus grande avec le système entraîné sur le SemCor qu'avec celui entraîné sur UFSAC. Ceci s'explique par la couverture du SemCor qui est moins importante que celle de tous les corpus UFSAC réunis. Pour le système appris sur le SemCor, la couverture est en effet de 91% sur SensEval 2, 97% sur SensEval 3, 93% sur SemEval 2007 (07), 98% sur SemEval 2007 (17), 91% sur SemEval 2013 et 95% sur SemEval 2015. Pour celui appris sur tout UFSAC, la couverture est respectivement de 98%, 99%, 99%, 99%, 98% et 99%.

Ces résultats démontrent la grande qualité du SemCor, c'est en effet lorsque sa couverture sur les tâches d'évaluation est la plus proche de 100% que notre système appris sur ce seul corpus obtient les meilleurs résultats. Les autres corpus UFSAC permettent quand même d'annoter un bien plus grand nombre de sens sans stratégie de repli, et nos meilleurs résultats sur SensEval 2, SemEval 2013 et SemEval 2015 sont obtenus avec le système appris sur tous les corpus réunis.

Le premier système de Yuan *et al.* (2016) obtient des résultats comparables aux nôtres mais comme nous l'avons souligné dans la section 2, le caractère privé de leur corpus d'entraînement contenant 100 milliards de mots pour leur modèle de langue rend très difficile la reproductibilité de leurs résultats.

Leur deuxième système (LSTM + LP) ajoute une étape de propagation de labels, dans laquelle ils augmentent automatiquement leurs données d'entraînement annotées en sens, en recherchant dans une grande quantité de textes non annotés des phrases similaires aux phrases annotées, et en portant les labels de sens depuis les phrases annotées, vers les phrases non annotées. Cette méthode apporte de meilleurs résultats sur la plupart des tâches, cependant ils récupèrent, pour leurs données non annotées, 1000 phrases prises aléatoirement sur le Web pour chaque lemme, sans plus de précisions, ce qui rend la reproductibilité des résultats encore plus difficile.

Le système de Raganato *et al.* (2017b) qui est quant à lui très semblable au nôtre obtient des résultats moins élevés malgré une plus grande complexité de leur modèle, et ils utilisent 2 couches de cellules LSTM bidirectionnelles de taille 2048 (1024 par direction), donc un total de 4096 unités cachées, ce qui est deux fois plus que notre modèle.

Pour leur second système (BLTM + att. + LEX + POS), les auteurs ont ajouté une couche d'attention à leur réseau, et ils effectuent de l'apprentissage multi-tâches, c'est à dire que leur réseau apprend à la fois à prédire un label de mot ou de sens, ainsi que la partie du discours (POS) du mot, et son label sémantique dans WordNet (LEX), la tâche est rendue ainsi plus complexe.

En comparaison avec ces autres systèmes, les nôtres obtiennent des scores supérieurs à ceux de Raganato *et al.* (2017b) dans la majorité des cas, malgré une complexité réduite au niveau de l'architecture. Nous obtenons des scores similaires ou légèrement inférieurs à ceux de Yuan *et al.* (2016) mais en utilisant largement moins de données pour l'apprentissage, et surtout des données librement accessibles.

Enfin, on voit que tous les systèmes supervisés à base de réseaux de neurones surpassent le système de Iacobacci *et al.* (2016) là où il a été évalué. Cette approche combinant des classifieurs linéaires de type *SVM* et des traits à base de vecteurs de mots obtenait pourtant des résultats état de l'art avant l'arrivée des systèmes neuronaux.

6 Vers une amélioration non supervisée

Dans cette section, nous présentons une première approche visant l'amélioration de notre système de manière complètement non supervisée, en s'appuyant sur des corpus non annotés en sens. Nous mettons ainsi en avant des pistes qui pourraient être approfondies dans de futurs travaux.

6.1 Approche

L'approche que nous avons suivie est en partie inspirée de la méthode de propagation de labels de Yuan *et al.* (2016), dans laquelle les auteurs transfèrent des annotations de sens de leur corpus manuellement annoté vers des phrases non annotées, pour étendre leurs données d'apprentissage.

Notre approche est aussi et surtout inspirée des méthodes d'apprentissage par transfert et apprentissage par mimétisme telles que Kim & Kim (2017); Buciluǎ *et al.* (2006); Hinton *et al.* (2015), dans lesquelles un ou plusieurs modèles "enseignant" vont transférer leurs connaissances à un modèle "élève" en lui montrant comment effectuer une tâche. L'élève va ainsi apprendre à recopier ce que font les enseignants, observer des exemples dans de nouveaux contextes et ainsi apprendre à mieux généraliser.

Dans le contexte de la DL et donc dans notre approche, les modèles enseignants sont des modèles capables d'annoter n'importe quelle séquence de mots en sens, et le modèle élève sera un nouveau modèle qui va être entraîné sur des données produites par les enseignants.

Plus particulièrement, nous avons utilisé comme modèle enseignant le système de DL qui a obtenu le meilleur score F1 (voir la Table 1) sur notre corpus de développement uniquement (SemEval 2015) afin d'éviter tout biais, c'est à dire celui entraîné sur toutes les données UFSAC avec la stratégie de repli.

Nous avons annoté avec ce système un million de phrases prises sur les données anglaises monolingues des campagnes d'évaluation de la traduction automatique WMT, et plus précisément le premier million de phrases du corpus "News Crawl 2016" accessible sur le site de la campagne d'évaluation WMT17 [9].

Ensuite, nous avons entraîné un nouveau modèle avec la même architecture sur ces données automatiquement annotées, en suivant le même protocole décrit dans la section 4, puis nous avons conservé l'ensemble de poids qui obtenait le meilleur score F1 sur le corpus de développement.

Enfin, nous avons poursuivi l'entraînement de ce modèle initialisé avec cet ensemble de poids mais cette fois ci sur les corpus UFSAC manuellement annotés, toujours pendant 20 *epochs* et en conservant le modèle avec le meilleur score sur le corpus de développement. Cependant pour cette dernière phase, le modèle a convergé très rapidement et obtenu ce meilleur score au bout d'environ une à deux *epoch*, ceci parce qu'il avait été pré-entraîné sur les données automatiquement annotées.

Nous avons réitéré cette dernière étape jusqu'à obtenir 8 modèles différents afin d'évaluer cette méthode, comme pour le système original, en moyennant les prédictions d'un ensemble de modèles.

9. http://data.statmt.org/wmt17/translation-task/news.2016.en.shuffled.gz

6.2 Résultats

Nous avons évalué le système “élève” sur les mêmes tâches que pour la section 5, avec et sans repli, et nous avons comparé ses scores avec ceux obtenus par le système “enseignant”, et avec le système état de l’art de Yuan *et al.* (2016). Les résultats sont dans la Table 2.

Système	SE2	SE3	SE07 (07)	SE07 (17)	SE13 (12)	SE15 (13)
Système “élève” (UFSAC + 1M News 2016)	73.03	68.48	84.12	60.95	68.57	*74.13
Système “élève” (UFSAC + 1M News 2016 + repli)	**74.23** (+0.48)	69.19 (-0.97)	**84.83** (+1.24)	**61.10** (+1.10)	**69.95** (+1.03)	***74.55** (+0.57)
Système “enseignant” (UFSAC + repli)	73.75	**70.16**	83.59	60.00	68.92	*73.98
Yuan *et al.* (2016) (LSTM)	73.6	69.2	82.8	64.2	67.0	72.1
Yuan *et al.* (2016) (LSTM + LP)	73.8	71.8	83.6	63.5	69.5	72.6

TABLE 2 – Scores F1 (%) obtenus par le système “élève” sur les tâches de DL des campagnes d’évaluation SensEval 2 (SE2), SensEval 3 (SE3), SemEval 2007 (SE07) tâches 07 et 17, SemEval 2013 (SE13) tâche 12 et SemEval 2015 (SE15) tâche 13. Les résultats préfixés par un astérisque (*) sont obtenus sur le corpus utilisé pour le développement pendant l’apprentissage. La différence entre le système élève (avec repli) et le système enseignant est affichée entre parenthèses. Le meilleur score entre l’élève et l’enseignant est affiché en gras, et le meilleur score de l’état de l’art est affiché en rouge.

Sur toutes les tâches d’évaluation, à part celle de SensEval 3, le système élève obtient ainsi des scores significativement supérieurs à ceux du système enseignant, et il obtient même des scores surpassant l’état de l’art sur les tâches de SensEval 2, SemEval 2007 (07), SemEval 2013 et SemEval 2015.

À travers ces résultats, on peut voir à quel point la mise en place de ce type d’apprentissage par transfert de connaissances peut s’avérer efficace pour la construction d’un système de DL robuste et de bonne qualité. Notre système ainsi entraîné obtient en effet des scores supérieurs à notre système original et à l’état de l’art sur la plupart des tâches d’évaluation, alors que nous avons uniquement utilisé comme ressource supplémentaire un million de phrases en anglais non annotées provenant d’un corpus en libre accès.

Cette approche est un premier pas pour l’amélioration du système de DL basé sur notre architecture neuronale sans utiliser de données annotées manuellement supplémentaires, et elle aide effectivement notre système à mieux généraliser, mais elle souffre encore de défauts évidents, en témoigne la baisse de résultats sur la tâche de SensEval 3.

Parmi les points que nous prévoyons d’améliorer nous souhaitons entre autres :

— une sélection plus fine des données à annoter par le système enseignant, plutôt que de prendre un million de phrases d’un corpus de *news* aléatoires, s’adapter au domaine de la tâche sur laquelle on souhaite s’évaluer ;

— une sélection des annotations produites par le système enseignant, pour éviter de reproduire les erreurs du modèle neuronal qui peuvent être facilement détectées, par exemple à l’aide d’une mesure de confiance basée sur sa couche de sortie.

7 Conclusion

Nous présentons dans cet article une nouvelle architecture de réseau neuronal pour la désambiguïsation lexicale à base de cellules *LSTM*. Les *LSTM* sont des cellules récurrentes largement utilisées dans les réseaux de neurones traitant des séquences tels que les systèmes *sequence-to-sequence* pour la traduction automatique ou les systèmes utilisant un modèle de langue prédisant la prochaine entrée d'une suite de mots. Notre modèle est composé d'une couche d'entrée qui prend une séquence de vecteurs de mots construits séparément, il a ensuite une couche cachée de cellules *LSTM* bidirectionnelles, et enfin il possède une couche de sortie entièrement connectée de la taille du nombre de sens possibles dans le dictionnaire utilisé. Ce modèle se distingue de ceux existants dans l'état de l'art par le fait qu'il permet d'annoter tous les mots d'une séquence donnée en une seule passe, contrairement à Yuan *et al.* (2016) et Kågebäck & Salomonsson (2016), pour lesquels chaque mot et chaque lemme est traité indépendamment. Il est aussi moins complexe et moins difficile à entraîner que celui de Raganato *et al.* (2017b).

Nous avons entraîné un système sur six corpus au format UFSAC (Vial *et al.*, 2017), à savoir le SemCor, le DSO, le WNGT, l'OMSTI, le MASC et l'Ontonotes, mais aussi un système sur le SemCor uniquement, et nous les avons évalués sur les tâches de DL des campagnes d'évaluation SensEval/SemEval. Les résultats montrent que nos systèmes obtiennent des scores équivalents à ceux des meilleurs systèmes neuronaux de l'état de l'art. Seul le système de Yuan *et al.* (2016) augmenté par les données issues de leur propagation de labels obtient des scores plus élevés. Cette augmentation indépendante de leur architecture neuronale est cependant basée sur l'utilisation de grandes quantités de textes pris aléatoirement sur le web, ce qui rend la reproductibilité difficile.

Nous avons ensuite présenté une amélioration de notre système à l'aide d'une approche par transfert de connaissances pour laquelle seulement un million de phrases initialement non annotées étaient ajoutées aux données d'entraînement afin d'obtenir un modèle plus robuste et performant. Nous avons présenté des résultats avec ce système qui surpassent significativement l'état de l'art sur toutes les tâches d'évaluation de la DL hormis deux, et nous avons proposé quelques pistes d'amélioration futures pour continuer dans cette voie.

Les études sur les systèmes à base de réseaux de neurones pour la désambiguïsation lexicale sont encore très récentes en atteste le faible nombre de systèmes existants pour le moment. C'est cependant une direction prometteuse, tant les résultats obtenus par ces nouveaux systèmes ont montré leur qualité sur les campagnes d'évaluation, dépassant les meilleurs systèmes non neuronaux. Dans le même temps, les récents travaux comme Raganato *et al.* (2017a) ou Vial *et al.* (2017) facilitent la création et l'évaluation rigoureuse de nouveaux systèmes de DL, étant donné que toutes les ressources annotées en sens *WordNet* sont disponibles librement et dans un format unifié.

Références

ANDOR D., ALBERTI C., WEISS D., SEVERYN A., PRESTA A., GANCHEV K., PETROV S. & COLLINS M. (2016). Globally normalized transition-based neural networks. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1 : Long Papers)*, p. 2442–2452, Berlin, Germany : Association for Computational Linguistics.

BOJANOWSKI P., GRAVE E., JOULIN A. & MIKOLOV T. (2017). Enriching word vectors with subword information. *Transactions of the Association of Computational Linguistics*, **5**, 135–146.

BUCILUǍ C., CARUANA R. & NICULESCU-MIZIL A. (2006). Model compression. In *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*, p. 535–541 : ACM.

CHAN W., JAITLY N., LE Q. V. & VINYALS O. (2016). Listen, attend and spell : A neural network for large vocabulary conversational speech recognition. In *ICASSP*.

CHAN Y. S., NG H. T. & ZHONG Z. (2007). Nus-pt : Exploiting parallel texts for word sense disambiguation in the english all-words tasks. In *Proceedings of the 4th International Workshop on Semantic Evaluations*, SemEval '07, p. 253–256, Stroudsburg, PA, USA : Association for Computational Linguistics.

CHO K., VAN MERRIENBOER B., BAHDANAU D. & BENGIO Y. (2014). On the properties of neural machine translation : Encoder–decoder approaches. In *Proceedings of SSST-8, Eighth Workshop on Syntax, Semantics and Structure in Statistical Translation*, p. 103–111 : Association for Computational Linguistics.

EDMONDS P. & COTTON S. (2001). Senseval-2 : Overview. In *The Proceedings of the Second International Workshop on Evaluating Word Sense Disambiguation Systems*, SENSEVAL '01, p. 1–5, Stroudsburg, PA, USA : Association for Computational Linguistics.

HINTON G., VINYALS O. & DEAN J. (2015). Distilling the knowledge in a neural network. *arXiv preprint arXiv :1503.02531*.

HOCHREITER S. & SCHMIDHUBER J. (1997). Long short-term memory. *Neural Computation*, **9**(8), 1735–1780.

HOVY E., MARCUS M., PALMER M., RAMSHAW L. & WEISCHEDEL R. (2006). Ontonotes : The 90In *Proceedings of the Human Language Technology Conference of the NAACL, Companion Volume : Short Papers*, NAACL-Short '06, p. 57–60, Stroudsburg, PA, USA : Association for Computational Linguistics.

IACOBACCI I., PILEHVAR M. T. & NAVIGLI R. (2016). Embeddings for word sense disambiguation : An evaluation study. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1 : Long Papers)*, p. 897–907, Berlin, Germany : Association for Computational Linguistics.

IDE N., BAKER C., FELLBAUM C., FILLMORE C. & PASSONNEAU R. (2008). Masc : the manually annotated sub-corpus of american english. In *Proceedings of the Sixth International Conference on Language Resources and Evaluation (LREC'08)*, Marrakech, Morocco : European Language Resources Association (ELRA). http ://www.lrec-conf.org/proceedings/lrec2008/.

KÅGEBÄCK M. & SALOMONSSON H. (2016). Word sense disambiguation using a bidirectional lstm. In *5th Workshop on Cognitive Aspects of the Lexicon (CogALex)* : Association for Computational Linguistics.

KIM S. W. & KIM H.-E. (2017). Transferring knowledge to smaller network with class-distance loss.

KINGMA D. P. & BA J. (2014). Adam : A method for stochastic optimization. *CoRR*, **abs/1412.6980**.

MIKOLOV T., SUTSKEVER I., CHEN K., CORRADO G. S. & DEAN J. (2013). Distributed representations of words and phrases and their compositionality. In C. BURGES, L. BOTTOU, M. WELLING, Z. GHAHRAMANI & K. WEINBERGER, Eds., *Advances in Neural Information Processing Systems 26*, p. 3111–3119. Curran Associates, Inc.

MILLER G. A. (1995). Wordnet : A lexical database. *ACM*, **Vol. 38**(No. 11), p. 1–41.

MILLER G. A., LEACOCK C., TENGI R. & BUNKER R. T. (1993). A semantic concordance. In *Proceedings of the workshop on Human Language Technology*, HLT '93, p. 303–308, Stroudsburg, PA, USA : Association for Computational Linguistics.

MORO A. & NAVIGLI R. (2015). Semeval-2015 task 13 : Multilingual all-words sense disambiguation and entity linking. In *Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015)*, p. 288–297, Denver, Colorado : Association for Computational Linguistics.

NAVIGLI R. (2009). Wsd : a survey. *ACM Computing Surveys*, **41**(2), 1–69.

NAVIGLI R., JURGENS D. & VANNELLA D. (2013). SemEval-2013 Task 12 : Multilingual Word Sense Disambiguation. In *Second Joint Conference on Lexical and Computational Semantics (*SEM), Volume 2 : Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013)*, p. 222–231.

NAVIGLI R., LITKOWSKI K. C. & HARGRAVES O. (2007). Semeval-2007 task 07 : Coarse-grained english all-words task. In *SemEval-2007*, p. 30–35, Prague, Czech Republic.

NG H. T. & LEE H. B. (1997). Dso corpus of sense-tagged english.

PENNINGTON J., SOCHER R. & MANNING C. D. (2014). Glove : Global vectors for word representation. In *Empirical Methods in Natural Language Processing (EMNLP)*, p. 1532–1543.

PRADHAN S. S., LOPER E., DLIGACH D. & PALMER M. (2007). Semeval-2007 task 17 : English lexical sample, srl and all words. In *Proceedings of the 4th International Workshop on Semantic Evaluations*, SemEval '07, p. 87–92, Stroudsburg, PA, USA : Association for Computational Linguistics.

RAGANATO A., CAMACHO-COLLADOS J. & NAVIGLI R. (2017a). Word sense disambiguation : A unified evaluation framework and empirical comparison. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics : Volume 1, Long Papers*, p. 99–110, Valencia, Spain : Association for Computational Linguistics.

RAGANATO A., DELLI BOVI C. & NAVIGLI R. (2017b). Neural sequence learning models for word sense disambiguation. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, p. 1167–1178 : Association for Computational Linguistics.

SNYDER B. & PALMER M. (2004). The english all-words task. In *Proceedings of SENSEVAL-3, the Third International Workshop on the Evaluation of Systems for the Semantic Analysis of Text*.

SRIVASTAVA N., HINTON G., KRIZHEVSKY A., SUTSKEVER I. & SALAKHUTDINOV R. (2014). Dropout : A simple way to prevent neural networks from overfitting. *J. Mach. Learn. Res.*, **15**(1), 1929–1958.

SUTSKEVER I., VINYALS O. & LE Q. V. (2014). Sequence to sequence learning with neural networks. In *Proceedings of the 27th International Conference on Neural Information Processing Systems - Volume 2*, NIPS'14, p. 3104–3112, Cambridge, MA, USA : MIT Press.

TAGHIPOUR K. & NG H. T. (2015). One million sense-tagged instances for word sense disambiguation and induction. In *Proceedings of the Nineteenth Conference on Computational Natural Language Learning*, p. 338–344, Beijing, China : Association for Computational Linguistics.

VIAL L., LECOUTEUX B. & SCHWAB D. (2017). *UFSAC : Unification of Sense Annotated Corpora and Tools*. Research report, UGA - Université Grenoble Alpes.

YUAN D., RICHARDSON J., DOHERTY R., EVANS C. & ALTENDORF E. (2016). Semi-supervised word sense disambiguation with neural models. In *COLING 2016*.

ZHONG Z. & NG H. T. (2010). It makes sense : A wide-coverage word sense disambiguation system for free text. In *Proceedings of the ACL 2010 System Demonstrations*, ACLDemos '10, p. 78–83, Stroudsburg, PA, USA : Association for Computational Linguistics.

Correction automatique d'attachements prépositionnels par utilisation de traits visuels

Sebastien Delecraz[1] Leonor Becerra-Bonache[2]
Benoit Favre[1] Alexis Nasr[1] Frederic Bechet[1]
(1) Aix-Marseille Univ, Université de Toulon, CNRS, LIS, UMR 7020, Marseille, France
(2) Univ Lyon, UJM-Saint-Etienne, CNRS, Laboratoire Hubert Curien, UMR 5516, Saint-Étienne, France
(1) prenom.nom@univ-amu.fr, (2) leonor.becerra@univ-st-etienne.fr

RÉSUMÉ
La désambiguïsation des rattachements prépositionnels est une tâche syntaxique qui demande des connaissances sémantiques, pouvant être extraites d'une image associée au texte traité. Nous présentons et analysons les difficultés de cette tâche pour laquelle nous construisons un système complet entraîné sur une version étendue des annotations du corpus *Flickr30k Entities*. Lorsque la sémantique lexicale n'est pas disponible, l'information visuelle apporte 3 % d'amélioration.

ABSTRACT
PP-attachement resolution using visual features

Resolving prepositionnal attachments is a syntactic task that requires semantic knowledge, which can be extracted from the composition of a picture associated with a text. We present and analyse the difficulties of performing this task, for which we build a full system trained on extended annotations from *Flickr30k Entities*. When lexical semantics are unavailable, visual information brings 3% improvement.

MOTS-CLÉS :
rattachement prépositionnel, analyse syntaxique multimodale, stratégie de correction.

KEYWORDS:
PP-attachment, multimodal parsing, correction strategy.

1 Introduction

Les langues naturelles sont intrinsèquement ambiguës. Une partie de ces ambiguïtés peut être levée en utilisant des indices présents dans la phrase, mais d'autres requièrent un accès au contexte dans lequel elles ont été produites. Pour reprendre l'exemple célèbre « *Jean regard l'homme avec un télescope* », l'ambiguïté du rattachement de la préposition *pour* pourrait facilement être levée si nous avions la possibilité de voir la scène.

Nous proposons dans cet article une méthode de résolution de rattachements prépositionnels fondée sur l'utilisation d'indices visuels. Pour cela, nous utilisons un corpus constitué de paires composées d'une photo et d'une légende décrivant cette dernière. Ce corpus a été annoté manuellement à différents niveaux. Au niveau de l'image, des rectangles (que nous appellerons boîtes), ont été identifiés et une catégorie sémantique a été associée à chacune d'entre elles. Au niveau du texte,

certains groupes nominaux ont été identifiés, ainsi que certains rattachements prépositionnels, pour un sous ensemble de prépositions fréquentes. De plus les boîtes correspondant à des groupes nominaux ont été appariées à ces derniers. Le fait de disposer simultanément de l'analyse de l'image (par l'intermédiaire des boîtes), et du texte, (à travers certains rattachements prépositionnels) ainsi que l'appariement entre boîtes et groupes nominaux permet d'établir un lien entre les deux modalités et d'utiliser des informations provenant de l'image pour traiter le texte.

Le système que nous proposons repose sur un détecteur d'erreurs de rattachement, proposant un rattachement alternatif s'il détecte une erreur. L'originalité de ce détecteur est qu'il permet de prendre en entrée des indices lexicaux, mais aussi visuels et conceptuels. Dans le groupe nominal *a ball in_front_of a dog with a red collar* (*une balle face_à un chien avec collier rouge*), par exemple, correspondant à l'image reproduite en Figure 1, la décision de rattacher *with* à *dog* plutôt qu'à *ball* peut se fonder sur des indices lexicaux évidents, mais pourrait aussi se fonder sur des indices visuels en étudiant, par exemple, les positions relatives des boîtes correspondant à *ball*, *dog* et *collar*.

La structure de l'article est la suivante. Nous dressons l'état de l'art du problème du rattachement prépositionnel dans la section 2, en se focalisant sur les études les plus pertinentes pour nos travaux. Notre corpus d'étude est présenté dans la section 3. La section 4 traite du problème de la détection des boîtes et de leur appariement avec des groupes nominaux. Le détecteur d'erreurs de rattachement est décrit dans la section 5 et la section 6 décrit les résultats obtenus sur notre corpus d'étude.

2 État de l'art

Le problème du rattachement prépositionnel a fait l'objet d'un grand nombre d'études en traitement automatique des langues. Il constitue un problème important et difficile pour les analyseurs syntaxiques. De nombreuses sources d'informations et méthodes ont été proposées pour le résoudre. Nous passons ici en revue les plus pertinentes pour nos travaux.

Deux sortes de ressources ont largement été utilisées dans la littérature pour résoudre le problème des rattachements prépositionnels : des bases de connaissances sémantiques (Agirre *et al.*, 2008; Dasigi *et al.*, 2017), et des corpus (Rakshit *et al.*, 2016; Mirroshandel & Nasr, 2016; Belinkov *et al.*, 2014; de Kok *et al.*, 2017). À notre connaissance, peu de travaux utilisent des informations multimodales pour traiter ce problème. Les travaux les plus pertinents pour nous sont ceux de Christie *et al.* (2016) ; leur approche consiste à réaliser simultanément l'analyse visuelle (identification de boîtes) et l'analyse syntaxique pour des paires (image, phrase) puis ils considèrent le produit cartésien des analyses syntaxiques et visuelles. Les différents paires se voient attribuer un score et la paire obtenant le meilleur score est alors sélectionnée. La différence principale entre nos travaux et les leurs est que nous produisons une unique analyse syntaxique et que cette dernière est corrigée en fonction des informations visuelles. De plus, nous menons des expériences sur un nombre de paires (images/légendes) beaucoup plus important (22800 contre 1822).

De nombreux travaux se sont intéressés à la mise en correspondance d'un segment de phrase et d'une partie d'image, pour différentes sortes d'applications, tel la génération de légende (Vinyals *et al.*, 2015; Fang *et al.*, 2015; Karpathy & Fei-Fei, 2015) et la recherche d'image (Coyne & Sproat, 2001; Chang *et al.*, 2015). Notre système réalise aussi l'alignement de segments de phrases (plus précisément des groupes nominaux) avec des boîtes dans l'image afin de pouvoir utiliser des caractéristiques multimodales, sans que cela soit notre objectif principal.

Nos travaux sont aussi en lien avec ceux sur l'apprentissage de relations visuelles. Les travaux les plus pertinents sont ceux de Peyre *et al.* (2017) dans lesquels les auteurs développent de nouveaux descripteurs visuels pour la représentation de relations entre les objets d'une image. Leur modèle repose sur des représentations multimodales des configurations d'objets pour chaque relation, et entraînent des classifieurs sur la relation des objets avec une supervision au niveau de l'image seulement (*i.e.* des annotations du niveau image comme *person on bike*, sans annoter les objets impliqués dans la relation). Alors que nous pourrions utiliser leurs classifieurs de relations spatiales, l'objectif de notre travail est différent. Nous nous intéressons au problème de la désambiguïsation des rattachements prépositionnels. Nous utilisons des caractéristiques visuelles similaires pour représenter la configuration spatiale des objets, mais les objets sont détectés et représentés d'une manière différente (en utilisant *YOLOv2* vs *Fast R-CNN*).

De nombreux chercheurs en psycholinguistique et en psychologie cognitive ont également étudié l'interaction entre la vision et le langage lors du traitement des phrases par l'humain (Spivey *et al.*, 2002; Coco & Keller, 2015). Ces travaux démontrent la pertinence de l'information visuelle pour les humains afin de résoudre l'ambiguïté linguistique. Cette information est également d'une grande importance au cours des premières étapes de l'acquisition du langage chez l'enfant, puisque la plupart des phrases reçues par les enfants sont liées à leur environnement visuel immédiat (Snow, 1972; Shaerlaekens, 1973). Même si les objectifs sont éloignés, nos travaux s'inspirent de ces idées.

3 Le corpus *Flickr30k Entities*

Il existe peu de corpus multimodaux (texte, image) qui associent des régions de l'image à des séquences de mots du texte. Dans cet article nous avons utilisé le corpus multimodal *Flickr30K Entities* (Plummer *et al.*, 2017) (*F30kE*) qui fournit ce type d'annotations et qui constitue une extension du corpus *Flickr30k* (Young *et al.*, 2014), une référence bien connue pour la description d'images par des phrases.

F30kE est composé de 32K images, chacune associée à cinq phrases la décrivant. Les chaînes de coréférences se rapportant aux mêmes entités sont annotées et liées aux boîtes englobantes des objets correspondant dans l'image (244K chaînes de coréférence et 276K boîtes sont fournies). De plus, chaque boîte est associée à une des catégories sémantiques suivantes : personnes, parties du corps, animaux, vêtements, instruments, véhicules, scène, et autres. Un exemple issu de ce corpus est reproduit dans la Figure 1.

Nous avons enrichi ce corpus avec une analyse syntaxique automatique des phrases, suivi d'une vérification manuelle des 29068 rattachements prépositionnels potentiellement ambigus du corpus. La correction de rattachement a été faite par un seul annotateur, qui avait à sa disposition uniquement la préposition cible dans la phrase et l'image.

4 Appariement automatique de boîtes et de groupes nominaux

Le modèle de correction que nous proposons suppose un appariement entre les boîtes détectées dans l'image et des groupes nominaux des légendes. Cette tâche se décompose en trois étapes : la détection des boîtes dans l'image, la détection des groupe nominaux dans la légende et, enfin, leur appariement.

FIGURE 1 – Exemple de l'annotation du corpus *F30kE*. L'image est décrite par 5 légendes, chacune annotée avec des entités. Les entités coréférentes à un élément visuel sont lié à la boîte correspondante.

Elles sont décrites successivement dans les trois sections suivantes et illustrés dans la Figure 2.

4.1 Détection des boîtes

La tâche de détection de boîtes dans une image consiste à prédire la présence ou l'absence d'un objet dans une image étant donné une liste d'objets que le système est en mesure de reconnaître. Lorsqu'un objet est reconnu, les coordonnées de la boîte qui le contient sont produites. Nous avons utilisé ici le modèle de détection d'objets temps-réel à base de réseau de neurones *YOLOv2* (Redmon & Farhadi, 2017) qui produit, pour une image donnée, une liste de boîtes.

Ce système se décompose de la façon suivante : il prend en entrée une image puis la découpe en grille. Pour chaque cellule de la grille le système prédit un nombre fixe de boîtes englobantes, un score de confiance pour chaque boîte, et une probabilité pour chaque catégorie. Les prédictions finales sont prises en multipliant les scores de confiance aux probabilité des catégories. Nous avons ré-entraîné le modèle *YOLOv2* sur le corpus *F30kE* en utilisant comme initialisation les poids fournis par les auteurs et en limitant le nombre de catégories aux huit catégories sémantiques du corpus *F30kE*. Seules les prédictions avec un score de confiance supérieur à 0.1 ont été retenues.

Sur les 14229 boîtes des images issues de notre corpus de test, le système en a détecté 7110 (un objet est considéré comme détecté si le score d'*Intersection over Union* (*IoU* : ratio entre l'aire de l'intersection et l'aire de l'union de deux boîtes) entre sa boîte de référence et la prédiction est supérieur à 0.5). Le détecteur atteint sur l'ensemble de test un rappel de 0.49 et une précision de 0.29. Si on prend en compte les catégories sémantiques, ces performances descendent respectivement à 0.25 et 0.15. Ces résultats nous montrent qu'il s'agit d'une tâche difficile et que le traitement automatique de l'image dans cette tâche de détection représente une première barrière à l'utilisation de l'information visuelle.

4.2 Détection des groupes nominaux

Bien que la détection de groupes nominaux soit une tâche largement étudiée, les groupes cibles dans notre travail correspondent à des objets visuels et peuvent différer par nature des groupes nominaux typiques issus d'une analyse syntaxique. Pour cette raison, le système est entraîné directement sur les

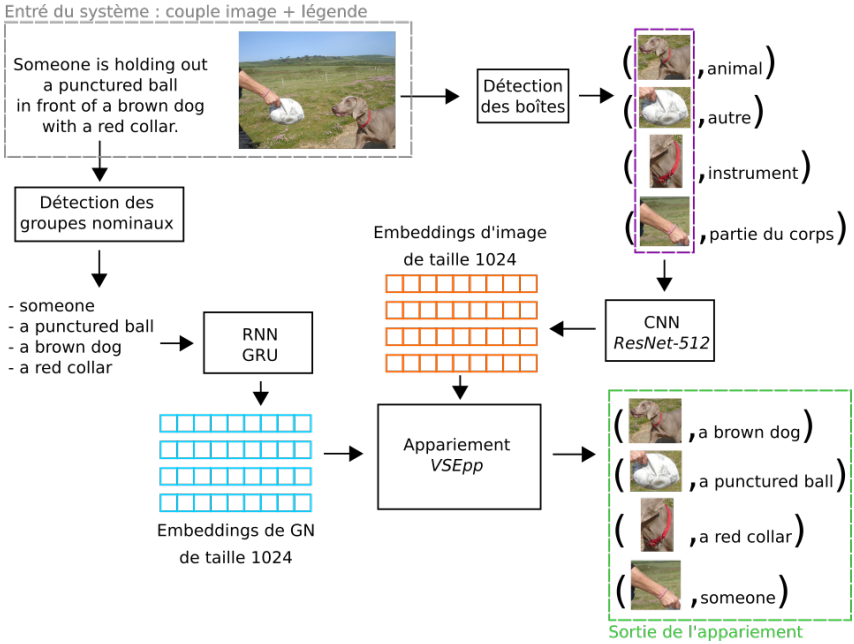


FIGURE 2 – Chaîne de traitement automatique pour l'appariement entre les boîtes des entités d'une image et les groupes nominaux de sa description.

groupes nominaux du corpus *F30kE*.

Il consiste en un simple détecteur de début et de fin de groupes nominaux, qui associe à tout mot de la phrase une étiquette de la forme B (*Begin*), I (*Inside*) et O (*Outside*) selon que le mot débute un groupe nominal, qu'il se trouve à l'intérieur d'un groupe nominal sans en être le premier mot ou qu'il se trouve à l'extérieur d'un groupe nominal. La prédiction est réalisée à l'aide d'un perceptron moyenné qui repose sur les mots de la phrase ainsi que leurs parties de discours. Une évaluation sur notre corpus de test indique un taux d'erreur de 2.2% par mot.

4.3 Appariement

Le problème de l'appariement consiste à déterminer pour chaque groupe nominal à quel objet détecté il correspond dans l'image. Par exemple, pour la légende de la Figure 2, il faut retrouver parmi les boîtes correspondant aux objets détectés (le ballon, le bras, le chien, le collier) à quels groupes nominaux ils correspondent (*someone, a punctured ball, a brown dog, a red collar*). C'est un problème de vision artificielle difficile du fait de la nature très différente des objets appariés : les pixels de l'image d'un côté et une séquence de mots de l'autre. Il est rendu encore plus difficile par le fait que certains groupes nominaux peuvent correspondre à plusieurs objets dans l'image (par exemple *children playing soccer*), certains objets ne sont représentés que partiellement dans la photo (*people standing in a train*), et le détecteur d'objet peut avoir détecté des objets non représentés dans la légende. Nous abordons cette tâche comme deux sous-tâches : la première consiste à calculer un score d'association entre chaque objet visuel et chaque groupe nominal, la seconde est de décider

parmi ces associations potentielles lesquelles seront conservées pour la suite. Nous ne traitons pas le problème des associations multiples.

Le score d'association entre un objet visuel et une séquence de mots est calculé en projetant les pixels de l'image et les mots de la légende vers un même espace de représentation. Chaque objet visuel et chaque séquence textuelle sont représentés par des vecteurs dans cet espace commun ce qui permet de calculer une similarité entre les vecteurs pour obtenir un score d'association. Cette projection dans un espace commun est réalisée à l'aide de réseaux de neurones. Les paramètres de ce réseau peuvent être entraînés à partir de paires connues $(\text{image}, \text{texte})$ selon la méthode décrite ci-après.

Cette méthode est fondée sur les plongements visuels sémantiques (*visual semantic embeddings*) de (Faghri *et al.*, 2017), qui tirent parti d'un réseau de neurones convolutionnel pour créer des représentations d'image et de réseaux de neurones récurrents pour créer des représentations des séquences de mots. Côté visuel, le contenu de chaque boîte est redimensionné en 224x224 pixels, puis passé en entrée à un réseau de type *ResNet-512* (He *et al.*, 2016) préentraîné sur la tâche ImageNet [1]. La dernière couche du réseau est remplacée par une couche dense (transformation linéaire) qui projette les représentations vers un vecteur de taille 1024. Côté texte, les mots des groupes nominaux sont d'abord projetés dans une couche d'*embedding* de taille 300 qui fournit des entrées à une couche récurrente de type GRU dont la représentation cachée est de taille 1024. La représentation cachée du réseau récurrent, à l'issue de la lecture des mots d'un groupe nominal est utilisée comme représentation pour la modalité textuelle. Les activations issues des réseaux de neurones pour les deux modalités sont normalisées par norme L2 et peuvent être comparées à l'aide du produit scalaire, ceci est équivalent au calcul de la similarité *cosine* entre les deux vecteurs.

L'apprentissage [2] des paramètres de ce modèle est effectué en calculant la similarité entre une paire $(\text{image}, \text{texte})$ existant dans les données d'apprentissage et une paire aléatoire avec l'un des deux membres en commun (*triplet ranking*), et en modifiant le modèle pour que le score de la paire valide soit supérieur à celui de la paire invalide (la fonction de coût utilisée est le *hinge loss*).

Méthode	Entraînement	Taux d'erreur
VSE++	F30k	42.07
VSE++	COCO	38.90
VSE	COCO	37.17
VSE	réentraîné	21.47

TABLE 1 – Taux d'erreur d'appariement sur notre ensemble de test en comparant les boîtes de référence et les groupes nominaux de référence, selon le modèle (VSE, VSE++) et le corpus d'entraînement (F30k et COCO sont les modèles fournis avec l'outil, entraînés sur des paires—phrase complète, image complète—plutôt que des groupes nominaux et le contenu des boîtes).

La Table 1 présente les performances du système d'appariement entre boîtes et groupes nominaux. Le taux d'erreur est calculé de la manière suivante : pour chaque groupe nominal, on considère que l'association est correcte si la boîte dont la similarité avec ce groupe nominal est la plus élevée est bien celle qui lui correspond dans les données de référence. Les résultats sont calculés selon deux méthodes, VSE et VSE++ qui diffèrent par la fonction de coût utilisée pour l'apprentissage (Faghri *et al.*, 2017) et selon le modèle utilisé (modèles fournis avec l'outil *VSEpp* ou modèle ré-entraîné

1. Reconnaissance de 1000 classes de scènes dans des images.

2. Implémentation basée sur `https://github.com/fartashf/vsepp`, mini-batches de taille 48, pendant 30 époques à l'aide de la méthode d'optimisation Adam.

sur les données de notre tâche). Les modèles disponibles avec l'implémentation *VSEpp* ont été entraînés sur des images complètes et des phrases de description complètes. Leurs performances s'écroulent sur les boîtes n'englobant qu'un objet dans notre corpus, doublant le nombre d'erreur d'association, comparé au même modèle ré-entraîné sur les données cible (21% $\rightarrow$ 37%) ce qui démontre l'importance de ré-entraîner le modèle dans des conditions identiques à celles du test.

Une fois le score d'appariement obtenu pour chaque paire $(\text{image}, \text{texte})$, il faut déterminer une association globale sachant qu'elle n'est ni injective ni surjective (certains éléments ne sont pas associés, d'autres ont des associations multiples). Cette association est réalisée à l'aide de l'heuristique suivante : les paires de plus fort score sont sélectionnées itérativement de manière gloutonne, chaque boîte pouvant être attribuée au plus à un groupe nominal. Seules les paires de score supérieur à 0.3 sont considérées (seuil déterminé sur un corpus de développement).

5 Détection d'erreurs de rattachement

La détection d'erreurs de rattachement est réalisée à l'aide du classifieur *Icsiboost* (Favre *et al.*, 2007) qui est fondé sur l'algorithme *Adaboost*. Pour entraîner ce classifier nous avons utilisé trois catégories de caractéristiques : lexicales, conceptuelles et visuelles. Ces caractéristiques portent sur la préposition p, son gouverneur G et son objet O. Lorsque le gouverneur est un verbe, c'est le sujet du verbe qui fait office de G. Ainsi, dans la phrase *Jean mange avec des gants.*, on obtient *G = Jean*, *p = avec* et *O = gants*.

En ce qui concerne les caractéristiques lexicales, en partant de l'arbre en dépendance produit par un analyseur nous utilisons : le lemme et la catégorie grammaticale du gouverneur et de l'objet, la distance entre la préposition et son gouverneur. Une description détaillée de ces caractéristiques est présentée dans de précédent travaux (Delecraz *et al.*, 2017).

Les caractéristiques conceptuelles et visuelles sont calculées à partir des boîtes englobantes que le système d'appariement a associé au gouverneur et à l'objet de la préposition. Les caractéristiques conceptuelles (personne, partie du corps, animal, vêtement, instrument, véhicule et autre) correspondent à celle utilisé dans le corpus *F30kE*, elles sont prédites lors de la détection d'objet dans l'image. Par ailleurs dans le cas où le module d'appariement n'a sélectionné aucune boîte pour un des deux groupes nominaux sélectionnés (gouverneur ou objet), la valeur *UNK* est utilisée pour représenter le concept de ce groupe nominal et aucune des caractéristiques spatiales n'est calculée.

En ce qui concerne les caractéristiques visuelles, étant donné deux boîtes $b_G = [x_g, y_g, w_g, h_g]$, $b_O = [x_d, y_d, w_d, h_d]$, où (x, y) sont les coordonnées du centre de la boîtes, et (w, h) sont la hauteur et la largeur de la boîte, nous utilisons les caractéristiques proposées par Peyre *et al.* (2017) :

$$V_{S1} = \frac{x_d - x_g}{\sqrt{w_g h_g}}\ , V_{S2} = \frac{y_d - y_g}{\sqrt{w_g h_g}}, V_{S3} = \sqrt{\frac{w_d h_d}{w_g h_g}}\ , V_{S4} = \frac{b_g \cap b_d}{b_g \cup b_d}, V_{S5} = \frac{w_g}{h_g}, V_{S6} = \frac{w_d}{h_d}$$

Les caractéristiques V_{S1} et V_{S2} représentent respectivement la position relative horizontale et verticale entre les centres deux boîtes. V_{S3} est le rapport entre les tailles des boîtes, V_{S4} l'*IoU* entre les deux boîtes, et V_{S5} et V_{S6} le rapport de forme des deux boîtes. L'annotation du corpus *F30kE* peut fournir plusieurs boîtes pour une même entité. Afin de faciliter nos calculs nous avons fait le choix de ne garder qu'une seule boîte par groupe nominal. Cette sélection se fait en gardant la boîte dont le centre est le plus proche du barycentre de toute les boîtes de l'entité.

En se basant sur toutes ces caractéristiques, le classifieur vérifie si le rattachement proposé par l'analyseur syntaxique est correct ou non. Afin d'augmenter la précision de l'analyseur syntaxique, nous utilisons une stratégie de correction qui consiste à changer le rattachement proposé par l'analyseur syntaxique en utilisant un correcteur d'erreur (Delecraz *et al.*, 2017). Lorsqu'un rattachement est détecté comme non correct par le classifieur, nous appliquons un ensemble de règles à l'arbre syntaxique généré par l'analyseur pour obtenir un ensemble de rattachements alternatifs. Ces nouveaux rattachements possibles sont donnés au classifieur pour prendre une décision finale en sélectionnant celui dont la probabilité de rattachement est la meilleure.

6 Cadre expérimental

Les expériences ont été réalisées à l'aide d'un analyseur en transitions entraîné sur le corpus *Penn Treebank* (Marcus *et al.*, 1993). Le corpus *F30kE*, enrichi des 29068 occurrences de préposition manuellement rattachées à leur gouverneur, a été subdivisé en trois ensembles : apprentissage (23254 prépositions), développement (2907 prépositions) et test (2907 prépositions). L'ensemble d'apprentissage a été utilisé pour entraîner le détecteur d'erreurs.

Les phrases de l'ensemble de test sont d'abord analysées puis les analyses produites sont fournies en entrée au détecteur d'erreurs qui propose éventuellement de modifier certains attachements prépositionnels. La Table 2 présente le taux de bon attachement pour les dix prépositions les plus courantes du test.

6.1 Résultats

Six configurations différentes sont évaluées : *Baseline* correspond au score obtenu par l'analyseur sans correction, V_C correspond au score obtenu après correction en n'utilisant que les caractéristiques conceptuelles correspondant aux neuf classes sémantiques distinguées dans le corpus. Dans la configuration V_S seules les six caractéristiques spatiales sont utilisées et dans V, c'est l'ensemble des caractéristiques visuelles (conceptuelles et spatiales) qui sont utilisées. Pour la configuration L, le correcteur utilise les caractéristiques linguistiques et, finalement, dans VL, c'est l'ensemble des caractéristiques qui sont prises en compte.

Comme nous pouvons le voir dans la Table 2, la précision initiale de l'analyseur syntaxique est de 75% sur les dix prépositions étudiées. Nous pouvons noter que les performances varient beaucoup selon les prépositions, variant de 95% pour la préposition *through*, à 33% pour la préposition *near*. L'utilisation de caractéristiques lexicales permet un gain absolu de 11%. L'apport des caractéristiques visuelles est plus modeste, il est de 3% avec, là aussi, une variabilité importante. Le gain semble surtout toucher des prépositions locatives comme *near* ou des prépositions qui ont un usage très différent du *Penn Treebank* comme *with*.

Trois raisons permettent d'expliquer les résultats modestes obtenus par les caractéristiques visuelles. Dans certains cas, il n'y a rien dans l'image qui permet de lever une ambiguïté d'attachement. D'autre part, dans les cas où l'information visuelle permet de lever l'ambiguïté, les caractéristiques utilisées ne représentent pas toujours les traits pertinents de l'image pour la lever. Finalement, les caractéristiques visuelles sont prédites. Cette prédiction est difficile : les boîtes ne sont pas toujours bien détectées et la classe sémantique qui leur correspond n'est pas toujours bien prédite. L'information lexicale

est elle beaucoup plus fiable, elle n'est pas prédite, mais donnée et, dans de nombreux cas, les mots suffisent à lever l'ambiguïté.

Lorsque l'on combine toutes les caractéristiques (visuelles et lexicales), il n'y a pas de gain en moyenne sur les prépositions sélectionnées, mais on observe des améliorations comme par exemple pour *into* qui dénote généralement des relations spatiales et pour laquelle le détecteur arrive à exploiter les caractéristiques visuelles.

Préposition	**#**	**Baseline**	V_C	V_S	V	L	VL
over	111	0.66	0.64	0.66	0.68	0.85	0.84
into	116	0.89	0.89	0.89	0.89	0.92	0.95
next to	137	0.89	0.89	0.89	0.89	0.90	0.89
from	140	0.76	0.76	0.76	0.76	0.86	0.85
on	143	0.85	0.85	0.85	0.85	0.89	0.87
through	145	0.95	0.95	0.95	0.95	0.96	0.96
near	159	0.33	0.53	0.50	0.59	0.84	0.84
for	168	0.73	0.73	0.73	0.72	0.82	0.83
with	310	0.65	0.68	0.68	0.70	0.78	0.78
in	369	0.76	0.76	0.77	0.76	0.85	0.84
TOTAL	1798	0.75	0.77	0.77	0.78	0.86	0.86

TABLE 2 – Taux de rattachements correct sur le test. # indique le nombre d'occurrences de la préposition ; La *baseline* est produite par l'analyseur syntaxique ; V_C représente les concepts visuels, V_S les caractéristiques spatiales, et L sont les caractéristiques lexicales.

6.2 Analyse d'erreurs

Nous présentons ici quelques exemples de couples $(\text{image}, \text{texte})$ pour lesquels l'image a permis, ou pas, de réaliser un rattachement correct en utilisant des ensembles de traits différents. Les Figure 3.a et 3.b montrent des phrases pour lesquelles l'analyseur a effectué un mauvais rattachement que le classifieur a permis de corriger en utilisant uniquement des informations visuelles. Dans la Figure 3.a, l'analyseur propose le mot *wall* comme gouverneur de la préposition *with*, et le classifieur corrige le rattachement en choisissant *sitting* comme gouverneur. Dans la Figure 3.b la préposition *near* est incorrectement rattaché à *area* par l'analyseur. Là encore, le classifieur permet de réaliser le rattachement correct. Ces exemples constituent la justification de cette étude : corriger de mauvais rattachements grâce à des informations visuelles.

Les Figures 4.a et 4.b montrent des phrases pour lesquelles l'utilisation de caractéristiques visuelles uniquement n'a pas permis de corriger un rattachement erroné. Dans la Figure 4.a, la préposition *on* est incorrectement rattachée au mot *building* et dans la Figure 4.b la préposition *in* est incorrectement rattachée au mot *crosswalk*. Dans les deux cas, le système d'appariement n'a pas trouvé de boîtes englobantes pour au moins un des deux groupes nominaux. Ces exemples constituent l'une des limites de cette étude : la difficulté de la phase de détection et d'appariement entre des boîtes et des groupes nominaux limite l'impact des traits visuels dans la correction des analyses erronées.

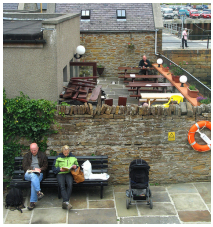
Même si les traits lexicaux sont les plus performants, si le corpus d'apprentissage ne contient pas assez d'exemples pour certaines entités, les caractéristiques visuelles peuvent s'avérer plus performantes. Ainsi les Figures 5.a et 5.b présentent des phrases pour lesquelles l'utilisation de caractéristiques

visuelles uniquement permet d'effectuer le bon rattachement, alors que l'utilisation de caractéristiques linguistiques seules produit une erreur. Dans la Figures 5.a, la préposition *with* est incorrectement rattaché au mot *jeans* à la place du mot *wearing*. Dans la Figures 5.b, la préposition *in* est rattaché au mot *bike* au lieu du mot *boy*.

a – A boy sitting on a concrete wall with a hat on.

b – Two children are in a grassy area near two horses.

FIGURE 3 – Exemples d'images pour lesquelles le classifieur a utilisé uniquement les caractéristiques visuelles pour corriger le rattachement de la préposition (en rouge).



a – Two people sitting in front of an older building on a bench.

b – Two younger women and an older man in a red sweatshirt are walking across a crosswalk in San Francisco.

FIGURE 4 – Exemples d'images pour lesquelles le classifieur a mal corrigé le rattachement de la préposition (en rouge) avec des caractéristiques visuelles uniquement.

a – A dog is wearing jeans and a blue and yellow shirt with a black vehicle in the background.

b – A boy with a black shirt and white shorts on a bike is turning to look behind himself.

FIGURE 5 – Exemples d'images avec un rattachement correct en utilisant des caractéristiques visuelles pour les prépositions en rouge alors que l'utilisation de traits lexicaux choisit un mauvais gouverneur.

7 Conclusions et perspectives

Ce travail explore la possibilité de tirer parti d'images pour désambiguïser les rattachements prépositionnels dans des phrases les décrivant, les caractéristiques visuelles améliorant en moyenne de 3 points les performances selon les prépositions, et parfois de manière drastique comme pour *near*. La difficulté du problème réside toutefois au niveau dans la détection et la catégorisation d'objets, ainsi que dans l'alignement entre le texte et les images.

Une meilleure utilisation de l'information provenant de l'image est une piste majeure d'amélioration du système avec notamment l'intégration de l'information issue directement des pixels (comme l'utilisation d'*embeddings* de l'image ou des boîtes englobantes).

Remerciements

Ces travaux ont été réalisés grâce au soutien financier apporté par la Direction Générale de l'Armement (DGA) en partenariat avec Aix-Marseille Université dans le cadre du *Club des partenaires Défense*. Les travaux de Leonor Becerra-Bonache ont été réalisés dans le cadre de sa délégation CNRS au Laboratoire d'Informatique et Système d'Aix-Marseille Université.

Références

AGIRRE E., BALDWIN T. & MARTINEZ D. (2008). Improving parsing and pp attachment performance with sense information. In *ACL*, p. 317–325.

BELINKOV Y., LEI T., BARZILAY R. & GLOBERSON A. (2014). Exploring compositional architectures and word vector representations for prepositional phrase attachment. *Transactions of the Association for Computational Linguistics*, **2**, 561–572.

CHANG A. X., MONROE W., SAVVA M., POTTS C. & MANNING C. D. (2015). Text to 3d scene generation with rich lexical grounding. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing of the Asian Federation of Natural Language Processing, ACL 2015*, p. 53–62.

CHRISTIE G., LADDHA A., AGRAWAL A., ANTOL S., GOYAL Y., KOCHERSBERGER K. & BATRA D. (2016). Resolving language and vision ambiguities together : Joint segmentation & prepositional attachment resolution in captioned scenes. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, p. 1493–1503.

COCO M. I. & KELLER F. (2015). The interaction of visual and linguistic saliency during syntactic ambiguity resolution. *The Quarterly Journal of Experimental Psychology*, **68**(1), 46 – 74.

COYNE R. & SPROAT R. (2001). Wordseye : an automatic text-to-scene conversion system. In *Proceedings of the 28th Annual Conference on Computer Graphics and Interactive Techniques, SIGGRAPH 2001*, p. 487–496.

DASIGI P., AMMAR W., DYER C. & HOVY E. (2017). Ontology-aware token embeddings for prepositional phrase attachment. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1 : Long Papers)*, volume 1, p. 2089–2098.

DE KOK D., MA J., DIMA C. & HINRICHS E. (2017). Pp attachment : Where do we stand ? *EACL 2017*, p. 311.

DELECRAZ S., NASR A., BECHET F. & FAVRE B. (2017). Correcting prepositional phrase attachments using multimodal corpora. In *Proceedings of the 15th International Conference on Parsing Technologies*, p. 72–77.

FAGHRI F., FLEET D. J., KIROS J. R. & FIDLER S. (2017). Vse++ : Improved visual-semantic embeddings. *arXiv preprint arXiv :1707.05612*.

FANG H., GUPTA S., IANDOLA F. N., SRIVASTAVA R. K., DENG L., DOLLÁR P., GAO J., HE X., MITCHELL M., PLATT J. C., ZITNICK C. L. & ZWEIG G. (2015). From captions to visual concepts and back. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2015*, p. 1473–1482.

FAVRE B., HAKKANI-TÜR D. & CUENDET S. (2007). Icsiboost. `http://code.google.com/p/icsiboost`.

HE K., ZHANG X., REN S. & SUN J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, p. 770–778.

KARPATHY A. & FEI-FEI L. (2015). Deep visual-semantic alignments for generating image descriptions. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2015*, p. 3128–3137.

MARCUS M. P., MARCINKIEWICZ M. A. & SANTORINI B. (1993). Building a large annotated corpus of english : The penn treebank. *Computational linguistics*, **19**(2), 313–330.

MIRROSHANDEL S. A. & NASR A. (2016). Integrating selectional constraints and subcategorization frames in a dependency parser. *Computational Linguistics*.

PEYRE J., LAPTEV I., SCHMID C. & SIVIC J. (2017). Weakly-supervised learning of visual relations. In *IEEE International Conference on Computer Vision, ICCV 2017*, p. 5189–5198.

PLUMMER B. A., WANG L., CERVANTES C. M., CAICEDO J. C., HOCKENMAIER J. & LAZEBNIK S. (2017). Flickr30k entities : Collecting region-to-phrase correspondences for richer image-to-sentence models. *International Journal of Computer Vision*, **123**(1), 74–93.

RAKSHIT G., SONTAKKE S., BHATTACHARYYA P. & HAFFARI G. (2016). Prepositional attachment disambiguation using bilingual parsing and alignments. *arXiv preprint arXiv :1603.08594*.

REDMON J. & FARHADI A. (2017). Yolo9000 : Better, faster, stronger. In *Computer Vision and Pattern Recognition (CVPR), 2017 IEEE Conference on*, p. 6517–6525 : IEEE.

SHAERLAEKENS A. (1973). *The Two-Word Sentence in Child Language Development : A Study Based on Evidence Provided by Dutch-Speaking Triplets*. The Hague : Mouton.

SNOW C. E. (1972). Mothers' speech to children learning language. *Child Development*, **43**(2), 549 – 565.

SPIVEY M. J., TANENHAUS M. K., EBERHARD K. M. & SEDIVY J. C. (2002). Eye movements and spoken language comprehension : Effects of visual context on syntactic ambiguity resolution. *Cognitive Psychology*, **45**(4), 447 – 481.

VINYALS O., TOSHEV A., BENGIO S. & ERHAN D. (2015). Show and tell : A neural image caption generator. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2015*, p. 3156–3164.

YOUNG P., LAI A., HODOSH M. & HOCKENMAIER J. (2014). From image descriptions to visual denotations : New similarity metrics for semantic inference over event descriptions. *Transactions of the Association for Computational Linguistics*, **2**, 67–78.

Décodeur neuronal pour la transcription de documents manuscrits anciens

Adeline GRANET[1], Emmanuel MORIN[1], Harold MOUCHÈRE[1],
Solen QUINIOU[1], Christian VIARD-GAUDIN[1]
(1) LS2N UMR 6004, 2 rue de la Houssinière, 44322 Nantes, France
prénom.nom@ls2n.fr

RÉSUMÉ

L'absence de données annotées peut être une difficulté majeure lorsque l'on s'intéresse à l'analyse de documents manuscrits anciens. Pour contourner cette difficulté, nous proposons de diviser le problème en deux, afin de pouvoir s'appuyer sur des données plus facilement accessibles. Dans cet article nous présentons la partie décodeur d'un encodeur-décodeur multimodal utilisant l'apprentissage par transfert de connaissances pour la transcription des titres de pièces de la Comédie Italienne. Le décodeur transforme un vecteur de n-grammes au niveau caractères en une séquence de caractères correspondant à un mot. L'apprentissage par transfert de connaissances est réalisé principalement à partir d'une nouvelle ressource inexploitée contemporaine à la Comédie-Italienne et thématiquement proche ; ainsi que d'autres ressources couvrant d'autres domaines, des langages différents et même des périodes différentes. Nous obtenons 97,27% de caractères bien reconnus sur les données de la Comédie-Italienne, ainsi que 86,57% de mots correctement générés malgré une couverture de 67,58% uniquement entre la Comédie-Italienne et l'ensemble d'apprentissage. Les expériences montrent qu'un tel système peut être une approche efficace dans le cadre d'apprentissage par transfert.

ABSTRACT

Neural decoder for the transcription of historical handwritten documents.

The lack of data can be an issue at the beginning of a study on new historical handwritten documents. To solve this issue, we present the decoder part of a multimodal approach based on transductive transfer learning for transcripting play titles of the Italian Comedy.

MOTS-CLÉS : modèle neuronal, apprentissage par transfert, transcription, Comédie Italienne.

KEYWORDS: neural model, transfer learning, transcription, Italian Comedy.

1 Introduction

La préservation de notre héritage culturel passe par la numérisation de documents historiques. La consultation de ces documents nécessite leur indexation afin de pouvoir y accéder efficacement. Pour cela, de nombreuses études interdisciplinaires faisant intervenir conjointement les sciences du traitement automatique des langues, de la reconnaissance des formes dans les documents et de la recherche d'information se sont développées.

Avec les documents anciens, l'un des enjeux importants concerne les évolutions orthographiques au fil du temps. Le problème de la normalisation de ces fluctuations est toujours délicat (Garrette

& Alpert-Abrams, 2016; Bollmann *et al.*, 2017). En reconnaissance d'objets, les enjeux sont plus nombreux et divers : détection et segmentation des lignes, détection automatique de mots-clés ou encore reconnaissance d'écriture. Ces dernières années, les compétitions tournant autour des documents historiques se multiplient (Cloppet *et al.*, 2016; Pratikakis *et al.*, 2016; Sanchez *et al.*, 2017). Les systèmes doivent s'adapter au support des documents, au niveau de détérioration ou encore au style de l'écriture. Tous ces facteurs ont un fort impact sur les systèmes.

En reconnaissance d'écriture, les réseaux les plus performants sont construits à partir de réseaux de neurones profonds, et plus récemment, ils intègrent des modèles d'attention comme (Bluche *et al.*, 2017). Les systèmes de reconnaissances sont construits avec des réseaux multi-dimensionnels utilisant des cellules de type *Long Short Term Memory* (MDLSTM), ou encore des réseaux récurrents à convolution (CRNN) associés à des réseaux bidirectionnels (BLSTM) (Granell *et al.*, 2018). Ces approches permettent d'utiliser tout le contexte disponible sur les images. Cependant, la phase de décodage de ces séquences dépend directement de la taille du vocabulaire utilisé pour construire un modèle de langage ou un dictionnaire. Lorsqu'une grande quantité de mots est hors-vocabulaire, les résultats se dégradent. Le problème majeur de ces réseaux est la quantité de ressources nécessaire à leur apprentissage.

L'étude envisagée concerne la Comédie Italienne pour laquelle aucune ressource annotée n'est disponible. Il n'est donc pas possible de mettre en œuvre directement un tel type de réseau.

L'apprentissage transductif par transfert de connaissance est une approche intéressante dans le cas où il y a un manque, voir une absence de données pour réaliser l'apprentissage d'un système. En effet, cette méthode consiste à utiliser différentes sources de données pour l'apprentissage d'un système dédié à une tâche, et appliquée sur des données différentes (Pan & Yang, 2010). Il est donc possible pour nous à partir de différentes données connues d'annoter des données inconnues. Ce procédé est utilisé pour alimenter les systèmes d'apprentissage gourmands dans différents domaines tels que la détection de mot automatique dans les documents historiques (Lladós *et al.*, 2012) ou pour les modèles multimodaux de traduction (Nakayama & Nishida, 2017). Notre solution est d'utiliser l'apprentissage par transfert de connaissances pour faire de la reconnaissance d'écriture sur ces nouveaux documents.

Les méthodes standards en traduction automatique utilisent des systèmes de type encodeur-décodeur à partir de réseaux neuronaux récurrents (Cho *et al.*, 2014a). Le premier élément encode une donnée source d'un langage en un vecteur, et le second élément décode la séquence dans une langue cible. Vinyals *et al.* (2015) a proposé un générateur de légendes pour les images constitué de deux réseaux : un réseau à convolution (CNN) pré-entraîné encodant une image dans un vecteur de taille fixe, et un réseau LSTM générant la description de l'image. Nous avons extrapolé cette approche pour l'appliquer à un système de reconnaissance d'écriture. La représentation intermédiaire retenue en sortie de l'encodeur utilise un espace explicite fondé sur les n-grammes.

Nous souhaitons en particulier mettre en place un système de type encodeur-décodeur afin de réaliser un apprentissage par transfert à deux niveaux : l'un pour encoder les images de mots et l'autre pour décoder vers le texte. Cette étude préliminaire vise à rendre compte de l'efficacité du décodeur à générer des mots issus du vocabulaire de la Comédie-Italienne de la représentation intermédiaire résultant de l'encodage de l'image.

2 Modèles pour l'apprentissage par transfert de connaissances

Nous souhaitons créer un système de reconnaissance d'écriture manuscrite pour des documents multilingues anciens à partir de ressources différentes en termes de langue et d'époque. En se basant sur les travaux réalisés pour la génération de description d'images, le modèle que nous proposons se décompose en deux parties complémentaires (voir Figure 1) comme (Vinyals *et al.*, 2015). La première partie a pour but d'encoder l'image d'un mot et de la convertir en un vecteur. La seconde partie, quant à elle, doit décoder ce vecteur pour en générer une séquence de caractères. L'originalité de notre approche réside dans le fait d'utiliser un vecteur de n-grammes comme pivot du système et de supprimer la notion de temporalité entre les caractères d'un mot. Ce vecteur permet d'encoder les informations dans un espace non-latent qui est transférable tant que les données d'apprentissage et les données de cible de transfert partagent le même alphabet. Une telle approche favorise un apprentissage indépendant des deux parties du système : le modèle optique et le modèle de langage.

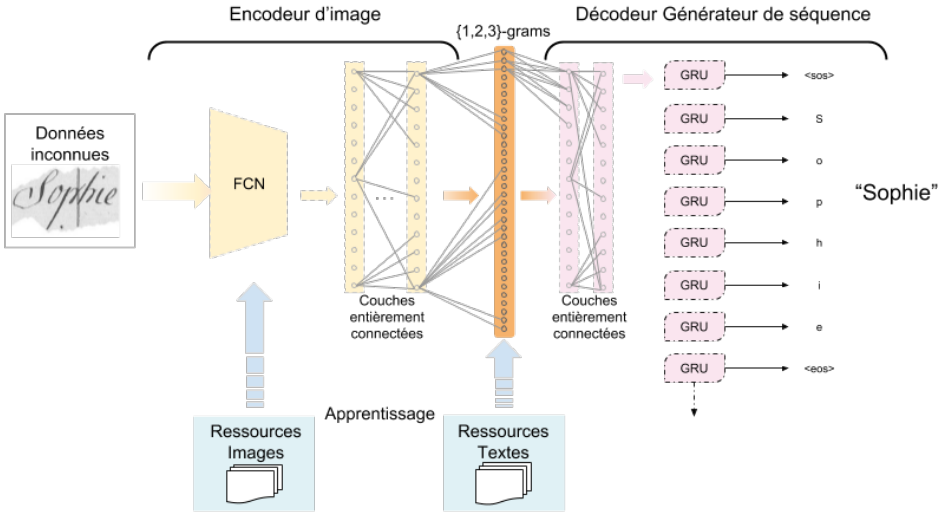


FIGURE 1 – Vue d'ensemble de l'architecture de l'encodeur-décodeur

2.1 Encodeur d'images

La tendance en apprentissage par transfert est à l'utilisation de réseaux pré-entraînés sur des images naturelles pour en extraire les caractéristiques. Ces images sont rarement en niveaux de gris contrairement aux images que nous exploitons. Nous avons donc choisi de définir et d'entraîner notre propre encodeur selon l'architecture suivante :

— un réseau entièrement à convolution (FCN) pour l'extraction des caractéristiques ;
— deux couches entières connectées avec une fonction d'activation de type ReLU (Nair & Hinton, 2010) et 1 024 neurones par couche ;
— une dernière couche entièrement connectée avec une fonction d'activation de type *Sigmoid* et $L + 1$ neurones où L correspond aux nombres de n-grammes estimés, et un neurone

supplémentaire comme joker si le n-gramme souhaité est absent de la liste.
Le FCN extrait et résume les caractéristiques des images prises en entrée. La dernière couche utilisant une *Sigmoid* permet d'obtenir une probabilité pour chaque n-gramme disponible et indépendamment des autres là où un *Softmax* normaliserait l'ensemble des sorties pour obtenir une somme égale à 1.

2.2 Décodeur de n-grammes

Dans le domaine du traitement du langage, il est courant d'utiliser des structures de type encodeur-décodeur prenant en entrée une séquence à partir de laquelle une autre séquence est générée, sans pourtant avoir la même longueur ou les mots dans le même ordre comme c'est le cas en traduction (Cho *et al.*, 2014b). Cette solution semble correspondre aux conditions imposées par l'apprentissage par transfert de connaissances et les données utilisées. Nous choisissons l'implémentation suivante :

— une première couche entière connectée avec une activation de type ReLU ;
— une couche récurrente de type GRU ;
— une dernière couche entièrement connectée avec une fonction d'activation de type *Softmax*.

Cette architecture volontairement simple pourrait l'être encore plus si nous utilisions une couche de plongement lexical pré-entraîné. Mais à notre connaissance, il n'existe pas de plongements lexicaux multilingues pour les n-grammes disponibles. Pour obtenir une séquence, nous utilisons une couche récurrente qui va générer de la temporalité. La dernière couche doit fournir un caractère jusqu'à ce que le symbole de fin de mot soit émis. Dans les modèles encodeur-décodeur notamment utilisés en traduction, la partie décodeur utilise un réseau bi-directionnel pour prendre en compte toutes les informations contenues dans une phrase à travers une matrice. Or, nous utilisons un vecteur comme entréece qui signifie que l'information temporelle spécifiant la position d'un n-gramme par rapport à un autre disparu. Un réseau bi-directionnel ne sera donc pas utile dans notre cas.

3 Ressources utilisées

Nous cherchons à réaliser une solution de reconnaissance d'écriture manuscrite sur les registres de la Comédie Italienne (RCI), pour lesquels nous avons peu de données annotées. Nous présentons les ressources utilisées pour nos expérimentations.

3.1 Données cibles : les registres de la Comédie Italienne

Ces documents sont des registres financiers de la Comédie-Italienne datant du XVIII^{e} siècle avec environ 28 000 pages. Voici quelques observations que nous avons pu effectuées :

— la langage évolue de l'Italien au Français, au début du siècle les acteurs eux-même rédigeaient les documents, avant d'engager un caissier ;
— la présentation des comptes journaliers change également au cours du siècle mais tout en préservant la présence des informations ;
— le style de l'écriture est très variable entre le début du siècle et 1730, avant de se stabiliser grâce au caissier.

La figure 2 présente un format de page d'un compte journalier où l'on distingue les informations suivantes : date du jour, titres des pièces qui ont été jouées ce soir-là, recettes et des notes (dans la colonne de gauche) et dépenses et liste des acteurs et actrices (dans la colonne de droite).

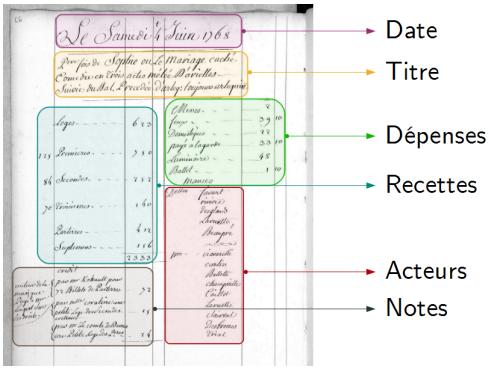


FIGURE 2 – Exemple d'une page journalière des registres financiers de la Comédie-Italienne avec l'identification des zones d'informations.

Dans la suite de nos expérimentations, nous nous concentrons sur la zone de titre. Cette zone contient une liste succincte des pièces jouées, qui peut être complétée par des informations indiquant si c'était une première de la pièce ou d'un acteur... La zone de titre de la figure 2 explique que la pièce « Sophie ou le mariage caché » a été jouée et que c'était une comédie en trois actes précédée des pièces « Arlequin toujours arlequin » et « mêlée d'ariettes ». Il faut également noter qu'un titre peut être écrit de plusieurs manières et reste principalement constitué d'entités nommées comme « Raton et Rosette » ou « Zémire et Azor ».

Pour le décodeur que nous réalisons, nous devons faire attention au langage et au style de l'écriture. En comparant le style contemporain de l'écriture avec l'historique, nous avons noté quelques différences dont la présence de caractères spéciaux comme la forme longue du 's' ; une évolution de la langue qui a converti les 'y' de fin de mot en 'i' ; ou encore le 'i' et 'j' qui ne se différenciaient pas.

Grâce à un site d'annotation participatif dédié à ces données de la Comédie Italienne, nous avons collecté 971 lignes de titres ce qui correspond à 4 938 mots au total. Nous utilisons l'ensemble des mots contenu dans ces lignes pour constituer la base de test pour le décodeur générateur de séquence, soit 1 431 mots différents. Cela reste un sous-ensemble du vocabulaire contenu dans l'ensemble des 28 000 pages de la Comédie-Italienne. Il est important pour l'apprentissage que nous sélectionnions des ressources avec des caractéristiques similaires à celles de la Comédie Italienne pour avoir un système efficace.

3.2 Les ressources pour l'apprentissage

Une nouvelle ressource. Pour alimenter le modèle de langage, nous pallions le manque de données de la ressource RCI en intégrant des données textuelles additionnelles provenant de 23 œuvres traitant de la Comédie Italienne, publiées au XVIII^{e} siècle et disponibles sur *Google Livre*. Parmi ces œuvres, nous trouvons des scripts bilingues (en Italien et en Français), des répertoires d'œuvres, des livres d'anecdotes sur le théâtre italien... Les œuvres obtenues ont été nettoyées pour supprimer le bruit induit par la numérisation, comme la structure du texte et les caractères spéciaux. Cette nouvelle ressource, appelé GCI, a l'avantage de posséder un vocabulaire proche des données manuscrites que nous étudions.

Les ressources existantes. Pour l'apprentissage par transfert de connaissances de l'encodeur d'images, nous avons sélectionné un ensemble de ressources manuscrites ayant au moins un point commun avec les données de la Comédie Italienne :

— RIMES (RM) (Reconnaissance et Indexation de données Manuscrites et de facsimilés) est une base française de demandes administratives (Grosicki & El-Abed, 2011) ;

— Los Esposalles (ESP) est construit sur des registres de mariages espagnols du XV^e au XVII^e siècle (Romero *et al.*, 2013) ;

— Georges Washington (GW) est une base en anglais créée à partir de 20 lettres de correspondance (Fischer *et al.*, 2012) ;

— Wikipedia français (Wiki), utilisée et distribuée par (Bojanowski *et al.*, 2017), est une ressource contenant tous les mots qui ont une fréquence supérieure à 5 dans Wikipedia. Nous en sélectionnons aléatoirement 30 000 mots.

	GCI	RM	ESP	GW	Wiki	RCI
Apprentissage	26 573	4 477	2 565	660	24 456	0
Validation	2 953	1,578	629	521	3 843	0
Test	0	1 627	629	431	1 928	1 431

TABLE 1 – Nombre de mots uniques des ressources exploitées.

La table 1 donne la taille du vocabulaire de chaque ressource. L'objectif est de transcrire les registres de la Comédie Italienne contenant 1 431 mots différents. Le taux de mots hors-vocabulaire par rapport aux autres ressources varie de 34 à 99 %. GCI obtient le taux le plus faible, ce qui confirme son choix, alors que le taux de mots hors-vocabulaire, par rapport à la ressource RM, est de 87,47 %.

Pour l'ensemble des ressources, nous avons remplacé les caractères accentués par leur forme simple, comme par exemple [é,è, ê,ë] par le caractère "e", ainsi que les formes spéciales comme la forme longue du "s", typique du XVIII^e, siècle en sa forme courte. Nous conservons également la casse. Pour les livres GCI, les lignes de texte ont été coupées sur les espaces et les signes de ponctuation. Pour supprimer les séquences de caractères qui ne sont pas des mots, nous avons uniquement conservé les séquences ayant une fréquence supérieure à 2 dans l'ensemble de la ressource.

Comme (Bengio & Heigold, 2014), nous utilisons les n-grammes de caractères pour représenter les mots. L'élément pivot de notre encodeur-décodeur est un vecteur de n-grammes de caractères. Initialement, les auteurs ont sélectionné les 50 k n-grammes les plus fréquents. Nous calculons tous les n-grammes possibles avec une longueur maximum de 3 sur l'ensemble des ressources d'apprentissage et en ajoutant les symboles de début *[* et fin de mot *]*. Par exemple, la décomposition du mot *Sophie* de la figure 1 est {[S,o,p,[So,op,Sop,...,ie],e]} soit un total de 19 n-grammes et d'une manière générale $3n + 1$ n-grammes de longueur maximale 3 pour un mot de n caractères. Pour GCI, nous filtrons les n-grammes ayant une occurrence strictement supérieure à 5 et ceux présents dans au moins deux ressources différentes, et un joker est ajouté pour remplacer les n-grammes non-sélectionnés. Il en résulte un nombre total de 12 500 n-grammes.

Pour l'entrée du décodeur, le vecteur de n-grammes est construit par normalisation de la fréquence de chaque n-gramme présent dans le mot. Cela permet de conserver une information sur la taille du mot et de compenser la séquentialité supprimée. Pour générer les séquences de sorties, nous avons 79 neurones représentant toutes les lettres minuscules, majuscules, les chiffres, les symboles de ponctuations dont l'espace et les symboles de début et fin de mot. Un dernier neurone est ajouté pour permettre au réseau de ne plus répondre de caractères après la fin du mot.

4 Paramétrage

Comme la taille de nos ressources varie de 660 à environ 25 000 mots, nous avons entraîné le décodeur avec une ou plusieurs ressources hormis pour GW (comme la taille est trop petite nous combinons systématiquement cette ressource à d'autres). Pour éviter le sur-apprentissage dans notre réseau, nous utilisons la méthode d'arrêt prématuré qui consiste à stopper le réseau quand au bout de cinq itérations la fonction coût sur la base de validation ne décroit plus. La structure du décodeur comprend 1 024 neurones dans la première couche entière connectée pour extraire les caractéristiques, suivi de 500 neurones cachés dans la couche GRU et enfin 79 neurones avec *Softmax* en fonction d'activation. Le taux d'apprentissage fixé à 0,0001 est géré automatiquement grâce à la fonction Adam. Pour pouvoir générer une séquence, nous fixons la longueur maximum à 50 caractères. Le réseau crée ainsi des séquences de longueurs différentes sans contrainte.

Nous évaluons notre système grâce à un taux de reconnaissance sur les caractères (TRC) et sur les mots (TRM). Le TRC est défini par $(N - (Ins + Subs + Dels))/N$ où N représente le nombre de caractères dans le mot de référence, $Subs$ le nombre de caractères substitués, $Dels$ le nombre de caractères supprimés et Ins le nombre de caractères insérés. Le TRM correspond au rappel, c'est-à-dire, le nombre de mots correctement reconnus par rapport au nombre de mots dans l'ensemble de référence. Nous calculons ces taux selon quatre options : i) avec et sans dictionnaire pour aider au décodage de la séquence et ii) avec et sans majuscule pendant le décodage. Le dictionnaire est construit sur le vocabulaire des ensembles d'apprentissage et de validation de l'ensemble des ressources. Cela nous donne un dictionnaire avec 39 051 entrées. Nous calculons également la couverture lexicale fournie par chaque ensemble d'apprentissage par rapport au test. Cela nous donne une borne haute pour le taux de reconnaissance de mots à atteindre avec le dictionnaire. Cette couverture lexicale correspond au nombre de mots communs à l'ensemble d'apprentissage et à celui de test, divisé par la taille du vocabulaire de l'ensemble de test.

5 Expériences

La table 2 montre les résultats obtenus pour la génération de séquences. Nous avons réalisé trois types d'expériences :

1. avec la même ressource pour l'apprentissage et le test ;
2. en ajoutant d'autres ressources pour l'apprentissage ;
3. en utilisant uniquement des ressources différentes de l'ensemble de test, pour l'apprentissage, ce qui correspond à l'apprentissage par transfert.

Même si le but de notre approche est bien de pouvoir décoder les mots de la Comédie-Italienne, nous présentons aussi les résultats obtenus sur RM et ESP. Cela nous semble intéressant de pouvoir observer si la méthode est applicable pour différents types de ressources.

Pour commencer, nous utilisons uniquement les unigrammes pour représenter un mot et la même ressource pour l'apprentissage et le test. Nous constatons avec RM, comme avec GCI, que les résultats sont meilleurs lorsque l'on considère les n-grammes de caractères. Sur GCI, le TRM augmente de 70 % sans l'utilisation du dictionnaire, tout en dépassant la couverture lexicale. Notons également que seul 3 % des caractères sont mal reconnus. Ces résultats corroborent les études utilisant les trigrammes comme (Vania & Lopez, 2017). Pour la suite des expériences, nous utilisons uniquement les vecteurs

de n-grammes.

Sur l'ensemble des expériences, nous obtenons globalement les mêmes résultats avec ou sans majuscules : $\pm$ 0.07 % pour le TRC, $\pm$ 1 % pour le TRM et $\pm$ 1.05 % avec le dictionnaire. Les erreurs de caractères du système ne sont donc pas uniquement des minuscules prédites en majuscules. L'utilisation du dictionnaire fait chuter les performances du décodeur. Nous remarquons que lorsque la couverture lexicale est supérieure à 20 %, le TRM reste inférieur à celle-ci. Le dictionnaire induit en erreur le décodeur quand il génère une séquence correcte et que le mot correspondant n'appartient pas au dictionnaire. Cependant, il est également capable d'aider le décodeur à se rapprocher du mot de référence même si la forme exacte n'est pas contenue dans le dictionnaire, et qu'en plus, les données d'apprentissage et de test utilisées sont d'époques et de langues différentes. Seulement, ces cas sont trop rares pour améliorer le TRC.

Test sur RCI Les résultats obtenus pour les TRC et TRM, lorsque GCI est combiné avec d'autres ressources, sont très similaires aux résultats obtenus sur GCI seul (Expérience 2 *vs.* Expérience 3, 4, et 5). Dans ce cas précis, l'augmentation de la quantité de données n'a pas un impact flagrant. Notons quand même que les meilleurs résultats sont obtenus en utilisant toutes les ressources en apprentissage. Dans le cas de l'apprentissage par transfert par rapport au domaine, la couverture lexicale est très basse, puisqu'elle est autour de 15 %. Cependant, le décodeur est capable d'atteindre 30,42 % de TRM en utilisant uniquement RM qui est en français contemporain, et 41,40 % en utilisant Wikipedia. Les résultats avec Wikipedia sont similaires à ceux obtenus en utilisant toutes les ressources avec une quantité inférieure de données. Finalement, chercher de nouvelles ressources encore inexploitées sur la Comédie Italienne est une approche intéressante : cela nous permet d'avoir un TRM supérieur de 20 % à la couverture lexicale. Parmi ces mots inconnus mais bien reconnus, nous retrouvons des abréviations telles que « arleq. » au lieu d'« Arlequin ».

Test sur RIMES Les résultats avec RM sont intéressants car c'est la seule ressource que nous utilisons qui est en français contemporain. Dans le cadre de l'apprentissage par transfert, sans RM dans les ressources d'apprentissage, les TRC et TRM atteignent les résultats obtenus sur RM seul. De plus, nous constatons qu'ils dépassent largement la couverture lexicale de 20 %. Lorsque nous travaillons avec des données historiques appliquées sur des données modernes mais partageant la même langue, la couverture lexicale est plus élevée que lorsque l'on utilise RM sur GCI. Ainsi, l'orthographe historique est plus facilement applicable sur du moderne pour le décodeur.

Test sur Los Esposalles Le vocabulaire de Los Esposalles est principalement construit à partir d'entités nommées. Ceci explique la couverture lexicale nulle dans le cadre du apprentissage par transfert. Cependant, le TRC est supérieur à 90 %, et le TRM dépasse la couverture lexicale de 52,91 %. Nous n'avons pas expérimenté l'utilisation seule de la base d'apprentissage de référence de cette ressource car elle est trop petite.

Analyse des erreurs La table 3 montre quelques erreurs récurrentes que nous avons pu constater. Parmi les erreurs observées, nous constatons que les mots ayant des caractères répétés, comme « cavalcade » et « clemence », posent plus de difficulté au système pour générer les caractères qui s'intercalent avec le caractère répété : il propose « cacaadade » et « ccceeene », respectivement. Cela représente environ 5% des erreurs constatées dans les expériences sur RCI. Une autre erreur commune

Test	Apprentissage	Expe. Id	N-grams	% Couv. lexicale	Sensible à la casse			Insensible à la casse		
					TRC	TRM	TRM dict.	TRC	TRM	TRM dict.
RCI	GCI	1	1	65,57	69,27	14,54	10,83	69,28	14,54	11,33
	GCI	2	1,2,3		97,10	86,22	39,30	97,17	86,26	40,14
	GCI+RM	3	1,2,3	67,58	**97,27**	**86,57**	**39,23**	97,27	86,57	40,07
	GCI+ESP	4	1,2,3	65,83	96,96	85,87	39,09	96,96	85,87	40,07
	GCI+ESP+GW+RM	5	1,2,3	67,65	95,85	79,65	38,25	97,42	87,13	39,16
	RM	6	1,2,3	14,52	79,70	30,42	17,27	79,75	30,49	17,76
	RM+ESP+GW	7	1,2,3	23,39	83,68	40,21	23,99	83,74	40,42	24,41
	Wiki	8.1	1,2,3	0.0	**87,32**	**41,40**	**25,24**	87,44	42,22	27,76
	Wiki 300k	8.2	1,2,3	0.0	92.80	55.94	29.37	93.00	57.27	31.61
RM	RM	9	1	75,09	83,97	43,07	28,49	83,98	43,07	28,98
	RM	10	1,2,3		94,72	79,50	37,78	94,74	79,63	37,84
	GCI+RM	11	1,2,3	83,83	98,25	92,0	40,49	98,25	92,0	40,55
	GCI+ESP+GW+RM	12	1,2,3	83,95	96,22	80,73	39,14	96,22	80,74	39,45
	GCI	13	1,2,3	58,55	**95,51**	**81,53**	**38,58**	95,51	81,53	38,58
	GCI+ESP	14	1,2,3	59,04	95,46	80,61	38,15	95,46	80,61	38,15
	Wiki	15	1,2,3	0.0	90,36	67,57	35,20	90,43	67,69	36,12
ESP	GCI+ESP	16	1,2,3	85,94	98,57	91,11	56,51	98,57	91,11	57,14
	GCI+ESP+GW+RM	17	1,2,3	86,10	98,40	90,79	57,62	98,40	90,79	57,78
	GCI	18	1,2,3	15,96	91,68	**65,87**	44,76	91,69	65,87	44,76
	RM	19	1,2,3	7,27	72,83	18,25	12,70	72,86	18,25	12,86
	GCI+RM	20	1,2,3	17,37	**92,05**	64,13	**46,51**	92,06	64,13	47,14
	GCI+RM+GW	21	1,2,3	17,69	91,68	64,60	44,60	91,70	64,76	45,07
	Wiki	22	1,2,3	0.0	84,52	34,28	32,38	84,71	35,08	34,12

TABLE 2 – Résultats pour la génération de séquence expérimentant l'apprentissage par transfert de connaissances : les TRC et TRM sont calculés avec ou sans l'utilisation du dictionnaire sur les différentes ressources.

est la permutation entre deux caractères comme avec « suitte » qui double un caractère à la place d'un autre. Cette erreur est la plus commune, elle couvre 79% des erreurs observées. Un dernier exemple avec « [ollat » pour « Soldat » qui apparaît lorsque le décodeur prédit deux symboles début de mot consécutifs, le second remplaçant le premier caractère du mot. Suivant la taille de la ressource utilisée pour l'apprentissage, ce type d'erreur représente entre 7% et 25% des erreurs, respectivement avec GCI, et RM seul.

Type d'Erreur	Expe. Id	Mot d'origine	Mot reconstitué
Caract. multiplié	4 3	cavalcade clemence	cacaadade ccceeene
Caract. interverti	6	suitte	usitte
Caract. de début	5 6	[diverstissemens] Soldat	ddevvestissemens] [ollat

TABLE 3 – Examples d'erreurs réalisées par le décodeur.

6 Conclusion

Dans cet article, nous nous sommes intéressés à la mise en place d'un apprentissage par transfert pour palier un manque de vérité terrain pour un système de reconnaissance d'écriture manuscrite. Contrairement aux travaux état-de-l'art, nous commençons par déconstruire la séquence initiale pour passer par une représentation intermédiaire robuste pour absorber les mots hors-vocabulaires (encodage), puis une séquence plausible est générée (décodage). Nos résultats montrent que l'approche est opérationnelle au niveau mot. Nous obtenons ainsi des TRC supérieurs à 90 % et des TRM dépassant la couverture lexicale estimée. Le décodeur que nous avons, est simple de part sa construction uniquement 4 couches mais nous obtenons de bons résultats. Cela renforce notre idée de rechercher de nouvelles ressources inexploitées au lieu de s'appuyer sur des ressources traditionnellement utilisées telles que Wikipedia. Dans le but de corriger les différentes erreurs que nous avons listées, des mécanismes d'attention pourraient être appliqués à plusieurs niveaux du système : en entrée de l'encodeur appliqué sur des images de ligne afin de se concentrer sur un mot à la fois, et générer un mot à la fois avec le décodeur ; dans le décodeur pour pouvoir prendre en compte les n-grammes qui ont pu être utilisé à chaque instant $t-1$ pour générer le nouveau caractère. Ces expériences ont été menées uniquement sur des mots et sans ponctuation. La prochaine étape sera donc d'évaluer ce décodeur sur des séquences beaucoup plus longues comme des lignes de titre contenant de la ponctuation.

7 Remerciements

Nous souhaiterions remercier les relecteurs pour leurs suggestions.

Références

BENGIO S. & HEIGOLD G. (2014). Word embeddings for speech recognition. In *Proceedings of the 15th Annual Conference of the International Speech Communication Association (Interspeech'14)*, Singapore.

BLUCHE T., LOURADOUR J. & MESSINA R. (2017). Scan, Attend and Read : End-to-End Handwritten Paragraph Recognition with MDLSTM Attention. In *Proceedings of the 14th International Conference on Document Analysis and Recognition (ICDAR'17)*, Kyoto, Japan.

BOJANOWSKI P., GRAVE E., JOULIN A. & MIKOLOV T. (2017). Enriching word vectors with subword information. *Transactions of the Association of Computational Linguistics*, **5**(1), 135–146.

BOLLMANN M., BINGEL J. & SØGAARD A. (2017). Learning attention for historical text normalization by learning to pronounce. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (ACL'17)*, p. 332–344, Vancouver, Canada.

CHO K., VAN MERRIENBOER B., BAHDANAU D. & BENGIO Y. (2014a). On the Properties of Neural Machine Translation : Encoder–Decoder Approaches. In *Proceedings of the 8th Workshop on Syntax, Semantics and Structure in Statistical Translation (SSST'14)*, p. 103–111, Doha, Qatar.

CHO K., VAN MERRIËNBOER B., GULCEHRE C., BAHDANAU D., BOUGARES F., SCHWENK H. & BENGIO Y. (2014b). Learning phrase representations using rnn encoder-decoder for statistical machine translation. *arXiv preprint arXiv :1406.1078*.

CLOPPET F., EGLIN V., KIEU V. C., STUTZMANN D. & VINCENT N. (2016). ICFHR2016 Competition on Classification of Medieval Handwritings in Latin Script. In *Proceedings of the 15th International Conference on Frontiers in Handwriting Recognition (ICFHR'16)*, p. 590–595, Shenzhen, China.

FISCHER A., KELLER A., FRINKEN V. & BUNKE H. (2012). Lexicon-free handwritten word spotting using character HMMs. *PRL*, **33**(7), 934–942.

GARRETTE D. & ALPERT-ABRAMS H. (2016). An unsupervised model of orthographic variation for historical document transcription. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics : Human Language Technologies (NAACL-HTL'16)*, p. 467–472, San Diego, CA, USA.

GRANELL E., CHAMMAS E., LIKFORMAN-SULEM L., MARTÍNEZ-HINAREJOS C.-D., MOKBEL C. & CÎRSTEA B.-I. (2018). Transcription of spanish historical handwritten documents with deep neural networks. *Journal of Imaging*, **4**(1), 15.

GROSICKI E. & EL-ABED H. (2011). ICDAR 2011 - French Handwriting Recognition Competition. In *Proceedings of the 11th International Conference on Document Analysis and Recognition (ICDAR'11)*, p. 1459–1463, Beijing, China.

LLADÓS J., RUSIÑOL M., FORNÉS A., FERNÁNDEZ D. & DUTTA A. (2012). On the influence of word representations for handwritten word spotting in historical documents. *IJPRAI*, **26**(05), 1263002–1–25.

NAIR V. & HINTON G. E. (2010). Rectified Linear Units Improve Restricted Boltzmann Machines. In *Proceedings of the 27th international conference on machine learning (ICML'10)*, p. 807–814, Haifa, Israel.

NAKAYAMA H. & NISHIDA N. (2017). Zero-resource machine translation by multimodal encoder–decoder network with multimedia pivot. *Machine Translation*, **31**(1-2), 49–64.

PAN S. J. & YANG Q. (2010). A survey on transfer learning. *IEEE Transactions on knowledge and data engineering*, **22**(10), 1345–1359.

PRATIKAKIS I., ZAGORIS K., BARLAS G. & GATOS B. (2016). ICFHR2016 Handwritten Document Image Binarization Contest (H-DIBCO 2016). In *Proceedings of the 15th International Conference on Frontiers in Handwriting Recognition (ICFHR'16)*, p. 619–623, Shenzhen, China.

ROMERO V., FORNÉS A., SERRANO N., SÁNCHEZ J. A., TOSELLI A. H., FRINKEN V., VIDAL E. & LLADÓS J. (2013). The ESPOSALLES database : An ancient marriage license corpus for off-line hwr. *PR*, **46**(6), 1658–1669.

SANCHEZ J. A., ROMERO V., TOSELLI A. H., VILLEGAS M. & VIDAL E. (2017). ICDAR2017 Competition on Handwritten Text Recognition on the READ Dataset. In *Proceedings of the 14th International Conference on Document Analysis and Recognition (ICDAR'17)*, p. 1383–1388, Kyoto, Japan.

VANIA C. & LOPEZ A. (2017). From Characters to Words to in Between : Do We Capture Morphology ? In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (ACL'17)*, p. 2016–2027, Vancouver, Canada.

VINYALS O., TOSHEV A., BENGIO S. & ERHAN D. (2015). Show and tell : A neural image caption generator. In *Computer Vision and Pattern Recognition (CVPR), 2015 IEEE Conference on*, p. 3156–3164 : IEEE.

Articles courts

A prototype dependency treebank for Breton

Francis M. Tyers[1] Vinit Ravishankar[2]
(1) School of Linguistics, Higher School of Economics, Moscow
(2) Institute of Formal and Applied Linguistics, Charles University in Prague, Prague
ftyers@hse.ru, vinit.ravishankar@gmail.com

Résumé

Cet article décrit le développement du premier corpus syntaxiquement annoté de breton. Le corpus fait partie du projet «Universal Dependencies». Dans cet article, nous décrivons la préparation du corpus, certaines constructions spécifiques au breton qui avaient besoin d'un traitement spécial et nous donnons des résultats de l'analyse syntaxique de breton par un nombre d'analyseurs syntaxiques.[1]

Abstract

A dependency treebank for Breton

This paper describes the development of the first syntactically-annotated corpus of Breton. The corpus is part of the Universal Dependencies project. In the paper we describe how the corpus was prepared, some Breton-specific constructions that required special treatment, and in addition we give results for parsing Breton using a number of off-the-shelf data-driven parsers.

Mots-clés : breton, analyse syntaxique de dependences, banque d'abres syntaxiques.

Keywords: breton, dependency parsing, treebank.

1 Introduction

Treebanks, or collections of sentences annotated according to some schema, have existed for decades, under a variety of standards intended to represent various details. The Universal Dependencies project (Nivre *et al.*, 2016) is a multilingual collection of treebanks annotated with dependency relations; at the time of writing of this paper, it consists of 115 treebanks in 65 languages. The project aims to enable a cross-linguistically valid schema of dependency annotation, and heavily depends on public contribution of mostly open resources. The existence of this unified collection of treebanks has led to extremely simplified parser creation and evaluation, as exemplified in the CoNLL 2017 shared task on dependency parsing (Zeman *et al.*, 2017).

This paper describes a treebank for Breton, a language spoken in Brittany in the north-west of France. The treebank will be included in the CoNLL 2018 shared task on dependency parsing[2], and we expect that it would provide a starting point for further annotation of Breton. The treebank is the second Celtic language treebank, and the first treebank for a language of the Brythonic subgroup of Celtic, in

[1] Berrskrid: Deskrivañ a ra ar pennad-mañ savidigezh ar c'horpus kentañ bet notennet e ereadurezh e brezhoneg. Ul lodenn eus ar raktres «Universal Dependencies» eo ar c'horpus-se. En teuliad e teskrivomp penaos e oa bet prientet ar c'horpus ha penaos e oa bet pledet gant frammoù dibar zo eus ar brezhoneg. Ouzhpenn-se, reiñ a reomp disoc'hoù dezrannadur ereadurel ar brezhoneg gant dezrannerioù ereadurel zo.

[2] http://universaldependencies.org/conll18/

Universal Dependencies.

The paper is laid out as follows, in Section 2 we give a brief sociolinguistic and typological overview of the Breton. Then in Section 3 we describe some prior work on computational resources and tools for Breton. In Section 4 we describe the composition of the corpus, and in Section 5 we describe some details of the annotation guidelines, paying attention to Breton-specific phenomena. Section 6.1 reports on a small experiment with three popular data-driven parsers, and is followed by some avenues for future work in Section 7 and conclusions in Section 8.

2 Breton

Breton (in Breton *brezhoneg*) is a Celtic language of the Brythonic branch which is today largely spoken in Brittany in the north-west of France. Historically it was spoken to different degrees throughout Brittany, but has been losing territory to French since the 12th century, most rapidly in the last 100 years. The language is classed as a language in "serious danger of extinction" by the *UNESCO Red Book on Endangered Languages* (Salminen, 1999). For an overview of Breton grammar, see Ternes (2008) and for full grammars see Press (1986) and Hemon (2007).

The language has two grammatical genders (masculine and feminine), two numbers (singular and plural)[3] and like the other Brythonic languages has lost the case system. Like other Celtic languages, Breton has contractions of pronouns and prepositions,[4] for example *ganin* 'with me' and *ganit* 'with you' (from *gant* 'with'). Unlike other Celtic languages, Breton also has an indefinite article *un* 'a', and an analytic passive construction with the verb *bezañ* 'be'.[5] Breton also exhibits the initial consonant mutation typical to Celtic languages, and has a fusional morphological system.

Syntactically, Breton has flexible constituent order within the sentence; VSO, SVO and OVS are frequently used with VSO — the classic Celtic order — being the most prominent. Adjectives follow nouns while other modifiers (adjectives, determiners) precede them. Verbs inflect for person, number, tense and mood. Auxiliaries may follow or precede the main verb.

3 Related work

There has been very little work to date on natural language processing for Breton. Among the related articles we may find Tyers (2009) who use a morphological analyser and bilingual dictionary to generate training data for statistical machine translation, and Tyers (2010) who describe a free/open-source rule-based machine translation system for Breton to French. There has been more recent work by Poibeau (2014) on treating initial consonant mutations with finite-state transducers. There are a number of written grammars of Breton, we have particularly relied on Press (1986) and Hemon (2007). Whilst no Breton treebanks exist that we know of, there are two existing treebanks for one Celtic language, viz. Irish (Lynn & Foster, 2016; Lynn *et al.*, 2016).

[3] A relic of a dual appears in some words relating to body parts, e.g. *divskouarn* '[a pair of] ears'.

[4] Often referred to in the literature as '*inflected* or *pronominal* prepositions'.

[5] The Welsh construction using the verb *cael* 'get', e.g. *Cafodd y llyfr ei ddarllen gan Yann* (lit. 'The book got its reading by Yann')

Source	Description	Sentences	Tokens	Average length
Grammar	Grammar book examples	277	2,092	7.55
Bremaik	Magazine articles	211	3,283	15.56
OfisPublik	Administrative texts	177	2,119	11.97
Wikipedia	Encyclopaedic texts	136	1,935	14.23
Examples	Translation examples	65	404	6.22
Songs	Traditional songs	21	251	11.95
Total		887	10,084	11.25

Table 1: Composition of the syntactically-annotated corpus.

4 Corpus

The corpus is composed of texts from a variety of domains (see Table 1 for a breakdown of the composition of the corpus). All of the texts are available under a free/open licence and the resulting corpus is distributed under the terms of the Creative Commons CC-BY-SA licence. In addition to the Breton sentences, each sentence has a translation in French or English. These have been produced either by a human (in the case of the songs, administrative texts and grammar book examples) or by the Breton–French MT system (Tyers, 2010) in the case of the magazine articles and Wikipedia. The texts were chosen to try and cover a range of written domains and grammatical structures. Our final annotated sentences

4.1 Preprocessing

Preprocessing the corpus consists of running the text through the Breton morphological analyser[6] available from Apertium (Forcada *et al.*, 2011) and described in Tyers (2009). This analyser also analyses initial consonant mutations[7] performs tokenisation of multi-word units based on the longest match left-to-right. The morphological analyser returns all the possible morphological analyses for each word based on a lexicon of around 18,900 lexemes. After tokenisation and morphological analysis, the text is processed with a constraint-grammar (Bick & Didriksen, 2015) based disambiguator for Breton consisting of 288 rules which remove inappropriate analyses in context. This reduces the average number of analyses per word from around 1.95 to around 1.06.

The native format of the treebank is the VISL format (Bick & Didriksen, 2015). This is a text-based format where surface tokens are on one line, followed by analyses on the subsequent line. The reason for choosing this format was that it was more convenient for hand-annotation, and was the format that the morphological analyser and constraint grammar output. We apply a number of deterministic transformations to convert the VISL format to CoNLL-U and a longest-match set overlap algorithm to convert the tagset from the Breton-specific one to Universal Dependencies.[8]

[6] `https://svn.code.sf.net/p/apertium/svn/languages/apertium-bre`

[7] Initial consonant mutations are where a word changes the first consonant due to morphological or syntactic context, e.g. *kazh* 'cat', but *he c'hazh* 'her cat'.

[8] The conversion code will be released along with the treebank in the final version.

5 Annotation guidelines

The annotation guidelines are based on Universal Dependencies (Nivre *et al.*, 2016), an international collaborative project to make cross-linguistically consistent treebanks available for a wide variety of languages. The Breton treebank is based on version 2.0 of the guidelines which were published in December, 2016. We chose the UD scheme for the annotation as it provides ready-made recommendations on which to base annotation guidelines. This reduces the amount of time needed to develop bespoke annotation guidelines for a given language; where the existing *universal* guidelines are adequate, they can be imported wholesale into the language-specific guidelines.

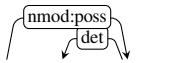
The treebank was annotated by a single human annotator; a translation was provided to aid the annotation process.

In the following subsections we describe some particular features of Breton that are interesting or novel with respect to the Universal Dependencies annotation scheme.

5.1 Nominal possessive construction

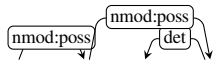
The possessive construction in Breton consists of a juxtaposition of a determined noun phrase with another noun phrase in the form where the non-determined noun phrase comes first (1a). Where there are more than two nouns, the determiner only comes between the two last nouns (1b). Both definite (1a, 1b) and indefinite (1c) determiners may fill the slot.

(1) a.



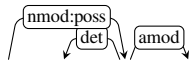
ti ar maer
NOUN DET NOUN
house the mayor
'the mayor's house'

b.



toenn ti ar maer
NOUN NOUN DET NOUN
roof house the mayor
'the roof of the mayor's house'

c.



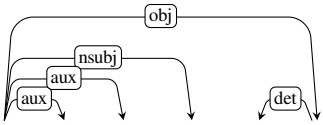
toenn un ti bras
NOUN DET NOUN ADJ
roof a house big
'the roof of a big house'.

The determiner slot can be also filled with possessive (e.g. *bugale ma c'hoar* 'my sister's children') and other determiners (e.g. *toenn pep ti* 'the roof of every house'). The use of `nmod:poss` is quite common across a variety of Universal Dependencies treebanks, including Irish, English and Persian.

5.2 Auxiliary items of verbs

There are two main auxiliary items of verbs in Breton, auxiliary verbs, such as *bezañ* 'to be' and *ober* 'to do'. These are used in forming the analytical tenses, such as the present and past (2a).

(2) a.

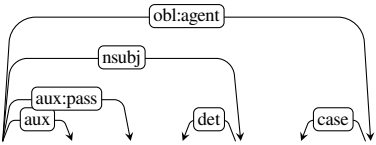


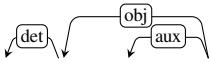
Lenn a reas Lenaig al levr
VERB PART AUX PROPN DET NOUN
Read did Lenaig the book
'Lenaig read the book'.

In common with other Celtic languages, Breton also has a number of verbal *particles* which serve a number of functions including negation and subordination. Both these and the auxiliary verbs are attached as dependents of the main verb (2a) with the relation `aux`. The use of this relation to mark auxiliary verbs is standard UD practice. The attachment of the particles to the main verb as opposed to the finite auxiliary may appear controversial. Grammars of Breton make the verbal particle subordinate to the finite verb. However as auxiliaries in UD may not have dependents, this leaves us with attaching these particles to the main verb.

5.3 Passive and impersonal

Unlike other Celtic languages, Breton has an analytic passive, made of the verb *bezañ* 'be' and the past participle of a transitive verb (3a). The agent in this construction may be omitted or expressed with a prepositional phrase using *gant* 'with'. We use the language specific relations `aux:pass` to mark the passive auxiliary and `obl:agent` to mark the demoted agent. Both subtypes are widely used cross-linguistically.





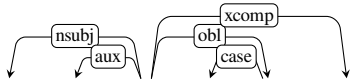
(3) a. Lennet e oa al levr gant Lenaig
VERB PART AUX DET NOUN ADP PROPN
read was the book by Lenaig
'The book was read by Lenaig'

b. Al levr a lenner.
DET NOUN PART VERB
the book was read
'The book was read.'

Breton also has an automonous (or impersonal) verbal form (3b), like in the other Celtic languages. In this construction the demoted agent cannot be expressed. In the Irish UD treebank (Lynn & Foster, 2016) the core arguments of these verbs are marked with the `obj` and we follow the same convention.

5.4 Contracted prepositions

Contractions of prepositions and pronouns (similar to the Spanish *contigo* 'with you') are widespread in the Celtic languages.[9] Unlike the Irish treebank (Lynn & Foster, 2016), which puts features indicating the person information of the contracted pronoun on the preposition, we use the two-level tokenisation scheme of UD to split them into a prepositional part and a pronominal part. Consider the sentence *Petra 'ra dit gouelañ ?* 'What is making you cry ?' (lit. 'What makes to-you crying?') in (4a), the contraction *da* + *it* = *dit* is split into a preposition *d-* and a pronoun *-it*.



(4) a. Petra ' ra d- -it gouelañ ?
PRON PART VERB ADP PRON VERB
What makes to you cry
'What is making you cry?'

[9] In the literature they are often called *inflected prepositions*, *conjugated prepositions*, *pronominal prepositions* or *prepositional pronouns*. As they do not have any special syntax we prefer the more cross-linguistic description of *contracted prepositions*.

System	Lemma	POS	Morph	UAS	LAS
Maltparser	-	-	-	73.80	65.82
BiST (MST)	-	-	-	74.78	67.00
UDPipe	88.24	89.21	87.02	75.24	68.14
UDPipe [+dict]	95.82	95.24	95.12	80.29	74.71
UDPipe [+dict,+embed]	95.80	95.15	95.00	79.85	74.29

Table 2: Performance of the systems on a number of tasks. Figures in brackets indicate equivalent scores on UD English, where available.

6 Experiments

6.1 Parsing performance

In order to test the treebank in a real setting, we evaluated three widely-used popular dependency parsers: Maltparser (Nivre *et al.*, 2007), UDPipe (Straka *et al.*, 2016) and BiST (Kiperwasser & Goldberg, 2016). In addition we provide results for using the treebank for part-of-speech tagging using UDPipe.

For Maltparser we used the default settings and for BiST parser we tested the MST algorithm.

We performed 10-fold cross-validation by randomising the order of sentences in the corpus, and splitting them into 10 equally-sized parts; in each iteration, we held out one part (90%) for testing (89 sentences) and used the rest (10%) for training (801 sentences). As the BiST parser required additional heldout data, we performed a 80-10-10 training-dev-test split per iteration. We calculated the labelled-attachment score (LAS) and unlabelled-attachment score (UAS) for each of the models using the CoNLL-2017 official evaluation script.[10] The same cross-validation splits were used for training all three parsers.

The morphological analyser and part-of-speech tagger in UDPipe was tested both with and without an external morphological dictionary. In this case the morphological dictionary, shown in Table 2 as `[+dict]`, consisted of a full-form list generated from the morphological analyser described in §4.1 numbering 296,905 entries. Further, the analyser was trained with dimension 300 fastText embeddings (Bojanowski *et al.*, 2016) `[+embed]`. These, unfortunately, did not improve our parsing results; we hypothesise that this was due to poor tokenisation of the embeddings training corpus (a generic 'language-independent' tokeniser will likely treat the Breton *c'h* letter incorrectly), and propose experiments on alternative forms of tokenisation for future work. Full results are presented in Table 2. Lemma, POS and morphology scores are absent for Maltparser and BiST as they are not included; each was evaluated on the output of a UDPipe instance (without embeddings and a dictionary). For comparison, a similarly set-up UDPipe instance (without external dictionaries or embeddings) achieves an LAS of 77.25 on English, 80.50 on French and 62.87 on Irish, which is likely the most comparable to our Breton treebank (Straka & Straková, 2017).

[10] `http://universaldependencies.org/conll17/evaluation.html`

7 Future work

The most obvious avenue for future work is to annotate more sentences. A treebank of 10,000 tokens is useful — it can be used for bootstrapping and also is key for evaluating unsupervised or semi-supervised systems — but in order to be able to train a parser useful for parsing unseen sentences we would need to increase the number of tokens 6–10-fold.

There are a number of quirks in the conversion process from VISL to CoNLL-U, for example the language-independent longest-common-subsequence algorithm could be replaced with a Breton-specific one that would be able to successfully split tokens like *en* (when it stands for 'in the') into *e* and *n* — the current generic algorithm gives *en* and *n*. We are also interested in collaborating with the authors of the Irish treebank to improve cross-linguistic compatibility.

8 Concluding remarks

We have described the first syntactically-annotated corpus of Breton. The treebank will be used as one of the languages in the 2018 CoNLL on dependency parsing and has been released for public use.[11] The corpus consists of a little over 10,000 tokens and is released under a free/open-source licence.

References

Bick E. & Didriksen T. (2015). Cg-3 – beyond classical constraint grammar. In *Proceedings of the 20th Nordic Conference of Computational Linguistics, NODALIDA*, p. 31–39: Linköping University Electronic Press, Linköpings universitet.

Bojanowski P., Grave E., Joulin A. & Mikolov T. (2016). Enriching word vectors with subword information. *arXiv preprint arXiv:1607.04606*.

Forcada M. L., Ginestí-Rosell M., Nordfalk J., O'Regan J., Ortiz-Rojas S., Pérez-Ortiz J. A., Sánchez-Martínez F., Ramírez-Sánchez G. & Tyers F. M. (2011). Apertium: a free/open-source platform for rule-based machine translation. *Machine Translation*, **25**(2), 127–144.

Hemon R. (2007). *Breton Grammar*. Evertype, 2nd edition.

Kiperwasser E. & Goldberg Y. (2016). Simple and accurate dependency parsing using bidirectional LSTM feature representations. *TACL*, **4**, 313–327.

Lynn T. *et al.* (2016). Irish dependency treebanking and parsing.

Lynn T. & Foster J. (2016). Universal Dependencies for Irish. In *Proceedings of CLTW 2016*.

Nivre J., de Marneffe M.-C., Ginter F., Goldberg Y., Hajič J., Manning C., McDonald R., Petrov S., Pyysalo S., Silveira N., Tsarfaty R. & Zeman D. (2016). Universal Dependencies v1: A Multilingual Treebank Collection. In *Proceedings of Language Resources and Evaluation Conference (LREC'16)*.

Nivre J., Hall J., Nilsson J., Chanev A., Eryigit G., Kübler S., Marinov S. & Marsi E. (2007). MaltParser: A language-independent system for data-driven dependency parsing. *Natural Language Engineering*, **13**(2), 95–135.

[11] https://github.com/UniversalDependencies/UD_Breton-KEB

Poibeau T. (2014). Processing mutations in Breton with finite-state transducers. In *Proceedings of the Celtic language technology workshop (CLTW) organised with COLING2014*.

Press I. J. (1986). *A Grammar of Modern Breton*. Mouton Grammar Library. Mouton.

Salminen T. (1999). *UNESCO Red Book on Endangered Languages*. UNESCO. `http://www.tooyoo.l.u-tokyo.ac.jp/archive/RedBook/index.html`.

Straka M., Hajič J. & Straková J. (2016). UDPipe: trainable pipeline for processing CoNLL-U files performing tokenization, morphological analysis, pos tagging and parsing. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, Paris, France: European Language Resources Association (ELRA).

Straka M. & Straková J. (2017). Tokenizing, pos tagging, lemmatizing and parsing ud 2.0 with udpipe. In *Proceedings of the CoNLL 2017 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies*, p. 88–99, Vancouver, Canada: Association for Computational Linguistics.

Ternes E. (2008). Breton. In D. MacAulay, Ed., *The Celtic Languages*, Cambridge Language Surveys. Cambridge University Press, 1 edition.

Tyers F. M. (2009). Rule-based augmentation of training data in Breton–French statistical machine translation. In *Proceedings of the 13th Annual Conference of the European Association of Machine Translation, EAMT09*, p. 213–218.

Tyers F. M. (2010). Rule-based Breton to French machine translation. In *Proceedings of the 14th Annual Conference of the European Association of Machine Translation, EAMT10*, p. 174–181.

Zeman D., Popel M., Straka M., Hajic J., Nivre J., Ginter F., Luotolahti J., Pyysalo S., Petrov S., Potthast M., Tyers F., Badmaeva E., Gökirmak M., Nedoluzhko A., Cinkova S., Hajic jr. J., Hlavacova J., Kettnerová V., Uresova Z., Kanerva J., Ojala S., Missilä A., Manning C. D., Schuster S., Reddy S., Taji D., Habash N., Leung H., de Marneffe M.-C., Sanguinetti M., Simi M., Kanayama H., dePaiva V., Droganova K., Martínez Alonso H., Çöltekin c., Sulubacak U., Uszkoreit H., Macketanz V., Burchardt A., Harris K., Marheinecke K., Rehm G., Kayadelen T., Attia M., Elkahky A., Yu Z., Pitler E., Lertpradit S., Mandl M., Kirchner J., Alcalde H. F., Strnadová J., Banerjee E., Manurung R., Stella A., Shimada A., Kwak S., Mendonca G., Lando T., Nitisaroj R. & Li J. (2017). Conll 2017 shared task: Multilingual parsing from raw text to Universal Dependencies. In *Proceedings of the CoNLL 2017 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies*, p. 1–19, Vancouver, Canada: Association for Computational Linguistics.

Détection automatique de phrases en domaine de spécialité en français

Arthur Boyer[1] Aurélie Névéol[1]
(1) LIMSI, CNRS, Université Paris-Saclay, rue John von Neumann, Campus Universitaire, F-91405 Orsay
prénom.nom@limsi.fr

RÉSUMÉ
La détection de frontières de phrase est généralement considéré comme un problème résolu. Cependant, les outils performant sur des textes en domaine général, ne le sont pas forcement sur des domaines spécialisés, ce qui peut engendrer des dégradations de performance des outils intervenant en aval dans une chaîne de traitement automatique s'appuyant sur des textes découpés en phrases. Dans cet article, nous évaluons 5 outils de segmentation en phrase sur 3 corpus issus de différent domaines. Nous ré-entrainerons l'un de ces outils sur un corpus de spécialité pour étudier l'adaptation en domaine. Notamment, nous utilisons un nouveau corpus biomédical annoté spécifiquement pour cette tâche. La detection de frontières de phrase à l'aide d'un modèle OpenNLP entraîné sur un corpus clinique offre une F-mesure de .73, contre .66 pour la version standard de l'outil.

ABSTRACT
Sentence boundary detection for specialized domains in French

Sentence boundary detection is generally considered as a solved problem. However, tools that perform well on standard text do not necessarily deal well with specialized corpus, which may degrade the analysis of other natural language processing tools intended to process sentence-segmented text. In this paper, we conduct a benchmark evaluation of 5 standard sentence boundary detection tools on 3 corpora covering different domains and subdomains. We then retrain one of the tools on domain-specific data and show that this leads to improved performance. In particular, we experiment with the clinical domain using a new clinical corpus annotated for gold-standard sentence boundaries. Sentence boundary detection with an openNLP model trained on the clinical data achieves an F-measure of .73, vs. .66 for standard openNLP distribution.

MOTS-CLÉS : Segmentation en phrases, domaine de spécialité, évaluation.

KEYWORDS: Sentence boundary detection, specialized corpus, benchmark evaluation.

1 Introduction

La segmentation en phrases, aussi appelée "détection de frontière de phrase" (DFP) ou "sentence boundary detection" en anglais, est l'une des premières étapes des chaînes de traitement du langage naturel, sur laquelle repose les étapes suivantes tel que la segmentation en mots (ou *tokenisation*), l'étiquetage morpho-syntaxique, la reconnaissance d'entités nommées. Les performances élevées obtenues pour les corpus journalistiques en anglais font que la segmentation en phrases est globalement considérée comme un problème résolu (Kiss & Strunk, 2006). Cependant, les bons résultats obtenus sur le domaine général ne se maintiennent pas toujours sur des domaines spécialisés, ce qui

a des répercutions sur l'ensemble des étapes postérieures dans une chaîne de traitement. De plus, les performances des outils de DFP varient probablement entre différentes langues à cause de différences morphologiques ou des ressources annotées disponibles. Nous nous sommes particulièrement intéressés à l'application d'outils de traitement automatique de la langue au domaine biomédical en français. À notre connaissance, il n'existe pas d'évaluation des outils ou méthodes de segmentation en phrase en français qui pourrait guider le choix des chercheurs selon les caractéristiques du corpus ou l'utilisation voulue. Pour combler ce vide, nous avons conduit une étude comparative de quatre outils de segmentation appliqués sur trois corpus en français.

2 Travaux proches

Dans les textes de domaine général et journalistique, le principal obstacle de la segmentation en phrases est lié aux abréviations. Les systèmes doivent identifier si un point indique une fin de phrase ou marque une abréviation, puis reconnaitre si cette marque d'abréviation est une fin de phrase (Gillick, 2009). De nombreuses méthodes reposent sur les signes de ponctuation comme le point de ponctuation, le point d'exclamation, le point d'interrogation, les points de suspension et les deux-points pour identifier les fins de phrase, soit avec un jeu de règles soit en apprenant à classer ces marqueurs à l'aide d'un corpus d'entrainement (Agarwal *et al.*, 2005; Urieli, 2013).

Les textes en domaine de spécialité apportent de nouveaux défis, touchant à la sémantique avec par exemple des abréviations absentes des dictionnaires non-spécialisés (comme les noms d'organismes tels que `E. Coli` ou `A. Thaliana` dans les textes biomédicaux) à la syntaxe (c'est-à-dire le manque de signes de ponctuation ou leur utilisation de façon non-conventionnelle) ou encore à la structure globale du document (abondance des listes à points, de titres de sections qui constituent des segments assimilables à une phrase). Les problèmes liés aux lettres capitales et aux ponctuations ont été longuement étudiés dans le contexte du traitement de l'oral et de la transcription de parole, où les signes de ponctuation ne sont pas disponibles (Treviso *et al.*, 2017).

Un autre domaine où les détections de frontières de phrases ont reçu une attention particulière est le domaine de la traduction automatique, qui s'appuie sur des corpus alignés au niveau des phrases pour l'entrainement de modèles statistiques. Quelques travaux ont évalué l'impact de performance de la segmentation dans la traduction automatique (Collados, 2013). D'autres travaux ont revisité la notion de "phrase" pour préparer les textes en segments plus courts, assimilables à des phrases (Kuang & Xiong, 2016).

Newman-Griffis *et al.* (2016) évaluent les outils de détection des frontières de phrases en anglais sur des corpus de domaines différents dont des textes journalistiques, des transcriptions d'appels téléphoniques, des résumés d'articles scientifiques et des textes cliniques. Dans leur travaux, ils mettent en avant les difficultés liées au traitement des textes cliniques avec des outils non-entrainés pour cette tâche spécifique. Miller *et al.* (2015) présentent des expériences sur des textes cliniques en anglais avec un modèle statistique entrainé sur des caractères et montrent que des résultats satisfaisant peuvent être obtenus avec une quantité limitée de données annotées. Kreuzthaler & Schulz (2015) ont abordé, avec des résultats positifs, le problème de la détection des abréviations et de la segmentation des phrases pour les textes cliniques en allemand.

3 Segmentation en phrases en français

3.1 Problématique

La segmentation en phrase est un problème qui est peu abordé en traitement automatique de la langue, car il est considéré comme résolu pour des textes de langue générale où les marqueurs de fin de phrase sont facilement identifiables et font partie d'une liste fermée restreinte. Pour d'autres types de texte issus des réseaux sociaux comme twitter, les travaux portent sur l'ensemble du segment de 140 caractères quil n'est pas nécessaire de redécouper. Cependant pour des textes de spécialité comme les textes du domaine biomédical, la question de la segmentation constitue un réel problème. En TAL les chaines de traitement commencent par une segmentation des textes en unités : des phrases, puis des syntagmes et des mots. C'est par exemple le cas de CTakes, un outil d'analyse des textes cliniques en anglais(Savova *et al.*, 2010), que nous souhaitons adapter à d'autres langues dont le français.

La segmentation en phrase présente plusieurs difficultés, d'ordre définitoire, méthodologique et technique. En effet, la définition d'une "phrase" est principalement accessible au travers des quelques corpus segmentés en phrases disponibles, comme le French Tree Bank (FTB) et Sequoïa. Par ailleurs, la rareté des corpus disponibles rend difficile l'évaluation de différentes méthodes et outils. En effet, la plupart des outils implémentant des méthodes d'apprentissage (par exemple, OpenNLP ou Talismane) sont entrainés sur le corpus FTB. Enfin, on constate également en pratique que la segmentation en phrases des outils s'accompagne d'une transformation du texte original : typiquement, Talismane propose une sortie au format coNLL tandis qu'OpenNLP présente le texte segmenté avec une phrase par ligne. Dans les deux cas des insertions, suppression ou substitutions de caractères (espaces ou ponctuation) posent des problèmes techniques supplémentaires pour l'alignement de deux versions d'un document, afin d'évaluer la segmentation proposée par rapport à une segmentation standard.

3.2 Contribution

Dans cet article nous proposons une contribution qui permet d'apporter des éléments de réponse à l'ensemble de ces difficultés. D'une part nous présentons deux nouveaux corpus annotés en phrases dans le domaine biomédical, ce qui permet de proposer une caractérisation de la phrase pour une variété de textes en français. Puis, nous nous appuyons sur ces nouvelles ressources pour faire des expérimentations sur la segmentation en phrases à l'aide des divers corpus et outils français disponibles. Nous présentons également un outil d'alignement de textes au niveau des phrases afin d'évaluer la segmentation.

3.3 Présentation des corpus et outils utilisés

Dans cette partie, nous présentons les corpus et outils utilisés pour réaliser notre étude. Le tableau 1 offre une description synoptique des corpus utilisés, que nous décrivons brièvement ci-dessous.

Afin d'élargir le nombre et la diversité des corpus français disposant d'une segmentation en phrase de référence, nous avons annoté deux corpus du domaine biomédical avec des frontières de phrases. Le **corpus EDP** est une collection de 338 titres et résumés d'articles biomédicaux. Il a été développé pour la tâche de traduction automatique dans le domaine biomédical dans le cadre de WMT 2017 (Jimeno Yepes *et al.*, 2017). Le **corpus MERLoT** (Campillos *et al.*, 2017) est un corpus composé de

Corpus	Type de Texte	Nombre de phrases	Long. moy. phrases
EDP	Articles scientifiques	3368	19.18
MERLoT	Textes Cliniques	7 836	6.34
French Treebank	Presse Nationale	21 564	24.41
Sequoia	Mixte	3 204	18.67
- Annodis	Presse Régionale	529	18.18
- EMEA	Notice de Médicament	1 118	16.39
- Europarl	Débat du Parlement Européen	561	23.24
- frwiki	Article d'encyclopédie	996	18.90

TABLE 1 – Description statistique des corpus utilisés.

documents cliniques désidentifiés issus des dossiers électroniques patient d'un groupe hospitalier français. Dans le cadre de ce travail sur la segmentation en phrases, nous avons annoté manuellement une partie du corpus (160 documents sur 500).Nous avons utilisé l'outil BRAT (Stenetorp *et al.*, 2012). Les documents ont été pré-annotés automatiquement en considérant comme fin de phrase chaque fin de ligne dans le corpus MERLoT, et chaque pontuation forte ou semi-forte pour le corpus EDP. Une série de six ensembles de cinq documents ont ensuite été corrigées par deux annotateurs, avec une réunion de consensus pour chaque série afin de discuter des désaccords et de finaliser le guide d'annotation. Une fois l'accord inter-annotateur stabilisé au-delà de .95 en F-mesure, chaque annotateur a travaillé indépendemment sur une partie des corpus restant.

Nous avons également utilisés les corpus existant **French TreeBank** (Abeillé *et al.*, 2003) pour l'entrainement d'outils statistiques, et le **corpus Sequoia** (Candito & Seddah, 2012) qui rassemble des textes issus de domaine différents. Pour cette étude, nous distinguerons le corpus médical EMEA (1 118 phrases) et les autres corpus : Annodis, Europal et frwiki, notés par la suite *Sequoïa-G* (2 086 phrases).

Les expériences de segmentation ont été réalisées avec quatre suites d'outils standard en TAL, dont une dédiée au français. Ces outils, que nous décrivons brièvement ci-dessous, ont été comparés à une baseline à base de règles qui marque une fin de phrase après les signes de ponctuation forts ".", " !", " ?" ou semi-forts " ;", " :" et les retours à la ligne.

- Stanford CoreNLP s'appuie sur des règles pour effectuer la segmentation en phrase à la suite de l'étape de tokenisation (Manning *et al.*, 2014).
- La suite OpenNLP d'Apache repose sur un classifieur MaxEnt pour la segmentation en phrases. Cet outil est intégré à la plateforme Ctakes. Pour notre travail nous l'avons entrainé sur le French TreeBank.
- L'analyseur Talismane (Urieli, 2013) met en oeuvre une segmentation en phrase par classification binaire d'une liste de signes de ponctuation ; entrainé sur le French Tree Bank.
- NLTK (Natural Language ToolKit) est un librairie python en open source. Le modèle français a été entrainé sur un corpus du journal *Le Monde*.
- Unitex (Paumier, 2016) est un outil d'analyse linguistique qui offre un outil de segmentation en phrases à base de règles (Friburger *et al.*, 2000).

Les performances des outils de segmentation ont été évaluées en termes de précision, rappel et F-mesure à l'aide d'un script permettant l'alignement du texte segmenté et de la référence au format une phrase par ligne.

4 Caractérisation des fins de phrase dans les corpus français

Afin d'illustrer les particularités des différents corpus du point de vue de la segmentation en phrases, le tableau 2 présente la distribution des marqueurs de fin de phrase observés. On constate que pour les corpus FTB ainsi que la partie non médicale du corpus Sequoïa, les marqueurs de fin de phrase sont majoritairement des pontuactions fortes. Les autres marqueurs sont des chiffres et lettres, indiquant la présence de phrases de type "titre". Les corpus médicaux (EDP, EMEA et Merlot) sont intermédiares et présentent une proportion importante de marqueurs de fin de phrase à l'aide de ponctuations semi-forte ou de marqueurs inhabituels.

Corpus	**Ponctuation forte** (. ? ! . . .)	**Ponctuation semi-forte** (; :)	**Chiffres et lettres**	**Autres marques**
FTB	90%	<1%	8%	1,6%
Sequoia	78,6%	3,2%	16,5%	1,6%
- non médical	84%	2,3%	12,9%	<1%
- médical (Emea)	68,4%	5%	23,3%	3,3%
EDP français	82,8%	16,9%	<1%	<1%
Merlot	27,3%	8%	25,3%	39,4%

TABLE 2 – Distribution des caractères de fin de phrase.

Par ailleurs, nous proposons ci-dessous quelques exemples représentatifs des cas difficiles que nous avons pu rencontrer dans les corpus médicaux. Dans chaque exemple, nous marquons la segmentation de référence par des crochets en gras, avec un numéro de segment en indice sur le crochet fermant. Dans (1) on observe une phrase contenant les caractères " :" et " ;" ne marquant pas une fin de phrase, alors que dans (2) les deux points marquent une fin de segment (titre) et le point virgule est utilisé comme une ponctuation forte marquant une fin de segment. (3) Illustre le cas de tableaux convertis. (4) illustre le cas d'une section de compte-rendu clinique rapportant des résultats d'analyse sous forme de liste non structurée.

(1) {Dans le cadre d'une dentisterie moderne, le praticien doit être à même d'apporter des solutions efficaces conjuguant : satisfaction du patient, en dissimulant ces défauts ; et économie tissulaire, avec l'approche la moins dommageable, laissant idéalement possible et aisée toute ré-intervention.}$_1$ **EDP - Actual. Odonto-Stomatol. 2014 ;269 :36-41**

(2) {Discussion :}$_1$ {La localisation et la teinte de la dyschromie concordaient avec la prise du traitement ;}$_2$ {tout ceci était étayé par l'absence de coloration chez la sœur jumelle.}$_3$ **EDP - Med Buccale Chir Buccale 2014 ;20 :279-283**

(3) {|IR DANS LE PARENCHYME |}$_1$ {POLE SUP |}$_2$ {MEDIAN |}$_3$ {POLE INF |}$_4$ {POLE SUP |}$_5$ {MEDIAN |}$_6$ {POLE INF |}$_7$
{| |0,78 |}$_8$ {0,81 |}$_9$ {0,79 | |}$_{10}$ {0,82 |}$_{11}$ {0,82 | }$_{12}$ **Extrait du corpus MERLoT**

(4) {EXAMENS COMPLEMENTAIRES : }$_1$ {Biologie : GB : 5,9 g/l. }$_2$ {PN : 3,0 g/l. }$_3$ {Plaquettes : 177 g/l. }$_4$ {Hb : 14,7 g/dl. }$_5$ {Créat. : 8,0.}$_6$ **Extrait du corpus MERLoT**

	Sequoia-EMEA			Sequoia-G			EDP			MERLoT		
	P	R	F	P	R	F	P	R	F	P	R	F
Stanford	.72	.49	.58	.87	.74	.80	.74	.81	.77	.66	.19	.29
OpenNLP	.78	.52	.63	.90	.83	.87	.81	.75	.78	.72	.61	.66
NLTK	.77	.53	.63	.91	.84	.87	.81	.74	.77	.56	.66	.61
Talismane	.78	.52	.63	.91	.84	.88	.81	.74	.77	.76	.62	.68
Unitex	.54	.73	.62	.75	.69	.72	.75	.81	.78	.63	.77	.69
Baseline	.68	.59	.63	.63	.81	.72	.92	.95	.93	.64	.68	.65

TABLE 3 – Evaluation d'outils de détection de phrases en français. Les performances sont mesurées en termes de précision (P), rappel (R) et F-mesure (F).

5 Expérimentations en segmentation

Le tableau 3 présente le résultat de l'application des outils de détection de phrase disponibles pour le français sur nos corpus de travail. On constate que les meilleures performances sont obtenues sur le corpus non médical Sequoïa-G qui présente le plus de ressemblance avec le French TreeBank, sauf dans le cas de la baseline qui s'avère bien adaptée au corpus EDP. Ces résultat reflètent la nature des corpus caractérisée par la distribution des caractères de fins de phrase présentées dans le tableau 2. En effet, une baseline fondée sur les ponctuations fortes et semi-fortes est particulièrement adaptée pour le corpus EDP dans lequel 98% des fins de phrases sont marquées par ce type de ponctuation. Le corpus MERLoT est celui qui présente le plus de diversité de fins de phrases avec presque 40% de marqueurs inhabituels, ce qui explique le faible rappel pour un outil comme Stanford, et conduit à des performances médiocres. A l'inverse, l'outil Unitex, également à base de règles, offre une bonne couverture ce qui se traduit par un rappel élevé. Néanmoins la précision reste faible et la F-mesure globale est similaire à celle des autres outils.

Nous avons également réalisé une série d'expériences plus spécifiques aux corpus du domaine biomédical (tableau 4). Nous avons entrainé des modèles statistiques fondés sur le maximum d'entropie (implémenté dans l'outil OpenNLP) sur plusieurs configurations des corpus d'entrainement :

1. dans la première configuration (Test1), on cherche à construire un modèle le plus ciblé au corpus de test, c'est à dire qui utilise le maximum de document du même corpus. L'entrainement est effectué sur deux tiers du corpus et le test sur le tiers restant.
2. dans la deuxième configuration (Test 2), on cherche à construire un modèle qui utilise le plus gros volume de données d'entrainement disponible pour chaque corpus. L'entrainement est effectué sur deux tiers des trois corpus et le test sur chacun des tiers restant des corpus médicaux.
3. dans la troisième configuration (Test 3) on cherche à construire un modèle qui utilise le plus gros volume de données d'entrainement disponible pour chaque corpus, tout en s'assurant que le corpus cible représente au moins un tiers des données d'entrainement. Le corpus d'entrainement est constitué d'une proportion égale des trois corpus. La disparité de taille entre les corpus fait que pour la composition du corpus d'entrainement de MERLoT il a été nécessaire d'utiliser l'intégralité des corpus EMEA et EDP.

La taille de ces corpus d'entrainement explique l'écart entre les resultats du tableau 3 et reportés pour OpenNLP dans le tableau 4 (rappelons que le corpus d'entrainement French Tree Bank utilisé comporte 21 564 phrases, soit plus du double de notre plus gros corpus d'entrainement spécialisé

utilisé dans les expériences du tableau 3). Les deux dernières stratégies semblent particulièrement adaptées pour le corpus clinique MERLoT, malgré la petite taille des corpus d'entrainement en comparaison avec le French Tree Bank.

	Sequoia-EMEA T=373				**EDP** T=1 123				**MERLoT** T= 2 612			
	E	P	R	F	E	P	R	F	E	P	R	F
Test 1	745	.21	.42	.28	2 245	.25	.62	.36	5 224	.31	.74	.43
Test 2	8 214	.71	.44	.54	8 214	.75	.60	.67	8 214	.78	.67	.72
Test 3	2 235	.73	.44	.55	5 608	.75	.60	.67	9 710	.78	.68	.73

TABLE 4 – Evaluation de modèles fondés sur le maximum d'entropie (OpenNLP) entrainés sur des corpus médicaux français. La taille de chaque corpus d'entrainment (E) et de test (T) est indiquée en nombre de phrases. Les performances sont mesurées en termes de précision (P), rappel (R) et F-mesure (F).

6 Conclusion et perspectives

Une conclusion assez surprenante de cette étude est que la performance des outils de segmentation en phrase pour le français est globalement modeste, en particulier en comparaison avec les performances sur l'anglais qui se situent bien au delà de .90 de F-mesure pour des corpus de langue générale. Concernant la segmentation en phrases pour les textes du domaine biomédical, il semble que le développement d'outils dédiés soit à base de règles soit statistiques reposant sur des corpus du domaine soit indispensable. Dans la suite de ce travail, nous prévoyons d'expérimenter avec des modèles statistiques reposant sur une segmentation en caractères, et d'évaluer l'impact de la segmentation en phrases sur des tâches d'extraction d'information comme la reconnaissance d'entités nommées ou l'extraction de relations.

Remerciements

Nous remercions le Service d'Informatique Biomédicale (SIBM) ainsi que l'équipe CISMeF du CHU de Rouen qui nous ont permis d'utiliser le corpus LERUDI pour cette étude. Ce travail a bénéficié d'une aide de l'Agence Nationale de la Recherche portant la référence CABeRneT ANR-13-JS02-0009-01.

Références

ABEILLÉ A., CLÉMENT L. & TOUSSENEL F. (2003). *Building a Treebank for French*, In A. ABEILLÉ, Ed., *Treebanks : Building and Using Parsed Corpora*, p. 165–187. Springer Netherlands : Dordrecht.

AGARWAL N., FORD K. H. & SHNEIDER M. (2005). *Sentence Boundary Detection Using a MaxEnt Classifier*. Rapport interne, Natural Language Processing Group, Stanford University.

CAMPILLOS L., DELÉGER L., GROUIN C., HAMON T., LIGOZAT A.-L. & NÉVÉOL A. (2017). A french clinical corpus with comprehensive semantic annotations : development of the medical entity and relation limsi annotated text corpus (merlot). *Language Resources and Evaluation*.

CANDITO M. & SEDDAH D. (2012). Le corpus Sequoia : annotation syntaxique et exploitation pour l'adaptation d'analyseur par pont lexical. In *TALN 2012 - 19e conférence sur le Traitement Automatique des Langues Naturelles*, Grenoble, France.

COLLADOS J. C. (2013). Splitting complex sentences for natural language processing applications : Building a simplified spanish corpus. *Procedia - Social and Behavioral Sciences*, **95**(Supplement C), 464 – 472. Corpus Resources for Descriptive and Applied Studies. Current Challenges and Future Directions : Selected Papers from the 5th International Conference on Corpus Linguistics (CILC2013).

FRIBURGER N., DISTER A. & MAUREL D. (2000). Améliorer le découpage des phrases sous Intex. *Revue Informatique et Statistique dans les Sciences Humaines*, **36**(1-4), 181–200.

GILLICK D. (2009). Sentence boundary detection and the problem with the u.s. In *Proceedings of Human Language Technologies : The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics, Companion Volume : Short Papers*, NAACL-Short '09, p. 241–244, Stroudsburg, PA, USA : Association for Computational Linguistics.

JIMENO YEPES A., NEVEOL A., NEVES M., VERSPOOR K., BOJAR O., BOYER A., GROZEA C., HADDOW B., KITTNER M., LICHTBLAU Y., PECINA P., ROLLER R., ROSA R., SIU A., THOMAS P. & TRESCHER S. (2017). Findings of the wmt 2017 biomedical translation shared task. In *Proceedings of the Second Conference on Machine Translation, Volume 2 : Shared Task Papers*, p. 234–247, Copenhagen, Denmark : Association for Computational Linguistics.

KISS T. & STRUNK J. (2006). Unsupervised multilingual sentence boundary detection. *Computational Linguistics*, **32**(4), 485–525.

KREUZTHALER M. & SCHULZ S. (2015). Detection of sentence boundaries and abbreviations in clinical narratives. *BMC Medical Informatics and Decision Making*, **15**(2), S4.

KUANG S. & XIONG D. (2016). *Automatic Long Sentence Segmentation for Neural Machine Translation*, In C.-Y. LIN, N. XUE, D. ZHAO, X. HUANG & Y. FENG, Eds., *Natural Language Understanding and Intelligent Applications : 5th CCF Conference on Natural Language Processing and Chinese Computing, NLPCC 2016, and 24th International Conference on Computer Processing of Oriental Languages, ICCPOL 2016, Kunming, China, December 2–6, 2016, Proceedings*, p. 162–174. Springer International Publishing.

MANNING C. D., SURDEANU M., BAUER J., FINKEL J., BETHARD S. J. & MCCLOSKY D. (2014). The Stanford CoreNLP natural language processing toolkit. In *Association for Computational Linguistics (ACL) System Demonstrations*, p. 55–60.

MILLER T. A., FINAN S., DLIGACH D. & SAVOVA G. K. (2015). Robust sentence segmentation for clinical text. In *AMIA Annu Symp*.

NEWMAN-GRIFFIS D., SHIVADE C., FOSLER-LUSSIER E. & LAI A. (2016). A quantitative and qualitative evaluation of sentence boundary detection for the clinical domain. **2016**, 88–97.

PAUMIER S. (2016). UNITEX 3.1 Manuel d'utilisation. http ://unitexgramlab.org/releases/3.1/man/Unitex-GramLab-3.1-usermanual-fr.pdf. Université de Marne la Vallée.

SAVOVA G., MASANZ J., OGREN P., ZHENG J., SOHN S., KIPPER-SCHULER K. & CHUTE C. (2010). Mayo clinical Text Analysis and Knowledge Extraction System (cTAKES) : architecture, component evaluation and applications. *J Am Med Inform Assoc*, **17**(5), 507–13.

STENETORP P., PYYSALO S., TOPIĆ G., OHTA T., ANANIADOU S. & TSUJII J. (2012). BRAT : a Web-based tool for NLP-assisted text annotation. In *Proceedings of the Demonstrations Session at EACL 2012*, p. 102–7.

TREVISO M. V., SHULBY C. & ALUÍSIO S. M. (2017). Sentence segmentation in narrative transcripts from neuropsychological tests using recurrent convolutional neural networks. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics, EACL 2017, Valencia, Spain, April 3-7, 2017, Volume 1 : Long Papers*, p. 315–325.

URIELI A. (2013). *Robust French syntax analysis : reconciling statistical methods and linguistic knowledge in the Talismane toolkit*. PhD thesis, Université de Toulouse II le Mirail.

Des représentations continues de mots pour l'analyse d'opinions en arabe : une étude qualitative

Amira Barhoumi[1, 2] Nathalie Camelin[1] Yannick Estève[1]
(1) LIUM, Le Mans, France - amira.barhoumi.etu@univ-lemans.fr , prenom.nom@univ-lemans.fr
(2) MIRACL, Sfax, Tunisie - amirabarhoumi29@gmail.com

RÉSUMÉ

Nous nous intéressons, dans cet article, à la détection d'opinions dans la langue arabe. Ces dernières années, l'utilisation de l'apprentissage profond a amélioré des performances de nombreux systèmes automatiques dans une grande variété de domaines (analyse d'images, reconnaissance de la parole, traduction automatique, ...) et également celui de l'analyse d'opinions en anglais. Ainsi, nous avons étudié l'apport de deux architectures (CNN et LSTM) dans notre cadre spécifique. Nous avons également testé et comparé plusieurs types de représentations continues de mots (*embeddings*) disponibles en langue arabe, qui ont permis d'obtenir de bons résultats. Nous avons analysé les erreurs de notre système et la pertinence de ces *embeddings*. Cette analyse mène à plusieurs perspectives intéressantes de travail, au sujet notamment de la constitution automatique de ressources expert et d'une construction pertinente des *embeddings* spécifiques à la tâche d'analyse d'opinions.

ABSTRACT

Word embeddings for Arabic sentiment analysis : a qualitative study

In this paper, we are interested in Arabic sentiment analysis task. Recently, the use of deep learning improves many automatic systems in a wide variety of fields (image analysis, speech recognition, machine translation, ...), among others English sentiment analysis. Thus, we study the performance of two architectures (CNN and LSTM) in our specific framework. In addition, we investigated the use of several types of word embeddings publically available for Arabic, that achieve good results. Finally, the analysis of the errors of our system and the relevance of the different embeddings was also proposed. These analysis lead to several interesting perspectives : building expert resources (lexicon) and relevant task-specific embeddings.

MOTS-CLÉS : Analyse d'opinion, représentation continue de mot, apprentissage profond, langue arabe.

KEYWORDS: Sentiment analysis, word embeddings, deep learning, arabic language.

1 Introduction

Avec la montée d'internet et la révolution des réseaux sociaux, un grand nombre d'individus peuvent exprimer leurs points de vue et leurs sentiments sur des entités, des produits, des personnes, *etc.* Dans ce contexte, le domaine de l'analyse automatique d'opinions connait un intérêt croissant de la part des entreprises et de la communauté scientifique [1]. Par ailleurs, les avancées scientifiques récentes dans les techniques d'apprentissage profond ainsi que la croissance des puissances de calcul, a mené à l'amélioration significative des performances dans différents domaines tels que la reconnaissance

1. https://trends.google.com/trends/explore?date=all&q=sentiment\%20analysis

de la parole ou la traduction automatique. La recherche en analyse d'opinions a également tiré profit de l'apprentissage profond, et plusieurs travaux ont été réalisés avec ce type d'apprentissage.

Dans cet article, nous nous focalisons sur la détection d'opinions par des méthodes à base de réseaux de neurones pour la langue arabe. Nous effectuerons nos experiences sur le corpus *Large-scale Arabic Book Review* (LABR) qui est un corpus de critiques de livres en langue arabe. Nous présentons en section 2 un état de l'art du domaine. Nous proposons ensuite, en section 3, nos deux systèmes neuronaux. Le premier s'appuie sur un réseau de neurones convolutifs CNN et le second sur un réseau neuronal réccurent de type *Long Short-Term Memory* LSTM. Nous étudions particulièrement l'utilisation de plusieurs types de représentations continues de mots disponibles pour la langue arabe (section 4). Nous analysons, en section 5, les erreurs de nos systèmes puis menons une analyse afin d'évaluer la pertinence des embeddings pour la tâche spécifique de détection d'opinions. Nous concluons et exposons les perspectives en section 6.

2 Etat de l'art

L'analyse d'opinions consiste à identifier la subjectivité et la polarité (positive, négative, neutre) d'un énoncé donné (Pang *et al.*, 2008). On peut l'appliquer au niveau du document, de la phrase ou d'un groupe de mots (Wilson *et al.*, 2004).

Les travaux effectués dans ce domaine peuvent être classés selon trois approches. La première est symbolique, elle utilise des lexiques et des règles linguistiques. La deuxième consiste en une approche statistique qui s'appuie sur des méthodes d'apprentissage automatique. Pour finir, il existe une approche hybride qui est une combinaison des deux précédentes : elle utilise à la fois des lexiques et des algorithmes d'apprentissage automatique. Jusqu'à récemment, les machines à vecteurs de supports SVM (Gaurangi *et al.*, 2014; Zainuddin & Selamat, 2014) et les classifieurs naïfs de Bayes NB (Tripathy *et al.*, 2015) représentaient les classifieurs les plus répandus dans ce domaine. Suivant la mouvance actuelle, les travaux récents font recours à l'apprentissage profond (Hassan, 2017; Deriu *et al.*, 2017; Zhou *et al.*, 2016).

Peu de travaux ont été réalisés pour l'analyse d'opinions en langue arabe. Ceci s'explique par le faible nombre de ressources développées et leur non disponibilité (Al-Kabi *et al.*, 2016). Nous citons quelques travaux existants selon leur catégorie. Suivant une approche linguistique, (Almas & Ahmad, 2007; Farra *et al.*, 2010) proposent une méthode s'appuyant sur un ensemble de patrons permettant d'extraire les polarités d'un document financier. Pour les travaux à base de lexiques, (Abdulla *et al.*, 2014a) construisent manuellement un lexique contenant 4815 mots. Leur système calcule le nombre de mots positifs et négatifs dans un texte afin de générer sa polarité globale. (Al-Kabi *et al.*, 2014) ont mis en place un outil qui détermine la subjectivité, la polarité d'une opinion et son intensité. Ils utilisent deux lexiques généraux et seize lexiques spécifiques. Suivant une approche statistique, (Abdulla *et al.*, 2014b) proposent un système de détection de subjectivité et de polarité dans les réseaux sociaux en utilisant des attributs morphologiques. (Bayoudhi *et al.*, 2015) comparent trois classifieurs : SVM, NB et un réseau de neurones simple. Pour finir, nous présentons les travaux à base de systèmes hybrides. (El-Halees, 2011) est le premier à avoir proposé un système hybride pour l'analyse d'opinions pour l'arabe. Il propose une hiérarchie séquentielle de classifications combinées. (Ibrahim *et al.*, 2015) utilise un lexique de 5244 adjectifs, un lexique de 3296 idiomes pour améliorer la classification de phrases avec un SVM. (Refaee & Rieser, 2016) appliquent une approche hybride pour la prédiction de l'intensité de la polarité dans les tweets. Ils ont utilisé particulièrement la régression logistique pour prédire les scores initiaux qui sont ajustés en appliquant des règles extraites à partir d'un lexique de polarité.

Plusieurs travaux récents appliquent des techniques d'apprentissage profond pour l'analyse d'opinion. (Barhoumi *et al.*, 2017) utilise les représentations continues de documents combinées avec un perceptron multicouche (PMC) tandis que (Dahou *et al.*, 2016) utilisent un CNN.

Nous détaillons dans la suite les différents systèmes que nous avons mis en place pour l'analyse d'opinions en arabe avec des méthodes d'apprentissage neuronal.

3 Systèmes d'analyse d'opinions pour l'arabe

Dans ce travail, nous nous intéressons à la classification selon leur polarité de critiques de livres en langue arabe. Nous avons implémenté deux systèmes : un CNN et un LSTM dont nous détaillons, dans la suite, les architectures. Nous décrivons également les différents types d'*embeddings* que nous avons utilisés.

3.1 Architectures à base de réseaux de neurones

Les réseaux convolutifs CNN ont prouvé leurs performances dans l'analyse d'opinions pour l'anglais (Kim, 2014). Nous avons donc choisi cette architecture pour implémenter notre premier système et évaluons ses performances pour l'arabe. Le CNN prend en entrée une matrice d'*embeddings* de taille fixe et applique une convolution de filtres, dont la taille de la fenêtre est une des valeurs de l'ensemble$\{3, 5, 7\}$, pour extraire de nouveaux attributs à partir de la matrice d'*embeddings*. Puis, un *max_pooling* est appliqué sur la sortie de la couche de convolution dans le but de conserver uniquement les attributs les plus pertinents qui sont concaténés au niveau d'une couche entièrement connectée. Enfin, le CNN applique la fonction *sigmoid* à la couche de sortie pour générer la polarité du document fourni en entrée. Deux polarités sont possibles : positif ou négatif (il s'agit d'une classification binaire).

Motivés par les bons résultats d'un système à base de réseaux LSTM pour l'anglais (Hassan, 2017), nous avons également décidé d'implémenter cette architecture. Il s'agit d'un cas particulier de réseaux de neurones récurrents (RNN) dont l'avantage principal est d'être composé d'unités neuronales appropriées pour permettre au réseau d'*oublier* ou de *mémoriser* : certaines observations du passé auront plus de poids que d'autres si elles sont jugées plus pertinentes pour la classification lors de l'apprentissage. Notre LSTM utilise comme entrée la même matrice d'*embeddings* que le CNN. Il est constitué d'une couche réccurrente de type LSTM unidirectionnelle simple connectée à une couche finale activée par une fonction *sigmoid*, pour générer la prédiction.

3.2 Représentations continues de mots arabes

Dans ce travail, nous avons utilisé deux ressources d'*embeddings* (disponibles gratuitement) comme entrée de nos systèmes neuronaux. La première ressource est celle de (Dahou *et al.*, 2016). Ils ont entrainé le modèle word2vec (Mikolov *et al.*, 2013) de type Skip-gram et *continuous bag of words* (CBOW) sur des pages web. Leurs expériences ont montré que CBOW est plus performant, ils l'ont donc mis à disposition. La deuxième ressource (Soliman *et al.*, 2017) est plus riche : elle regroupe six modèles d'*embeddings* entrainés sur trois types de corpus différents : twitter, wikipédia et des pages web. Ils ont entrainé CBOW et Skip-gram sur les trois types de corpus, mettant ainsi à disposition six

ensembles d'embeddings. Il est important de signaler que tous les embeddings disponibles sont de dimension 300.

4 Expériences

4.1 Corpus LABR

Pour évaluer nos systèmes, nous avons utilisé le corpus LABR (Nabil *et al.*, 2014) qui contient 63k critiques de livres composées d'un commentaire et d'une note associée (nombre d'étoiles). Nous nous plaçons dans le cadre d'une classification binaire et regroupons les critiques comme proposé dans (Nabil *et al.*, 2014) : les commentaires associés à une ou deux étoiles composent la classe *négative* et ceux à quatre ou cinq étoiles composent la classe *positive*. Ainsi les commentaires neutres ne sont pas considérés et le corpus utilisé se réduit à un ensemble de 40845 commentaires (68% positifs) pour le corpus d'apprentissage et 10211 pour le corpus de test (69% positifs). Notons que 10% de l'ensemble d'apprentissage est utilisé comme corpus de développement. Le corpus que nous utilisons est ainsi composé de 51k critiques, soit plus de trois millions de mots sur un vocabulaire de taille 324k. Pour mieux comprendre la distribution des mots, il est intéressant de connaitre les quelques statistiques suivantes : Le nombre d'occurences du mot le plus fréquent est de 76855 quand il est à 319 pour le 1000e mot le plus fréquent ; Si on considère qu'un mot peu fréquent est un mot qui apparait moins de 5 fois dans le corpus, on couvre alors 86,5% du corpus avec 13% du vocabulaire.

4.2 Comparaison des différents systèmes de détection d'opinions en arabe

Cette section expérimentale présente dans un premier temps les résultats récents des travaux déjà parus sur le corpus LABR. Nous notons que les meilleurs résultats ont été obtenus par (Dahou *et al.*, 2016) avec l'utilisation d'un CNN. Or, ces résultats n'ont pas été obtenus avec la répartition officielle du corpus. Le code des auteurs étant disponible, nous avons testé ce système sur la répartition officielle et avons obtenu 77,39% d'exactitude [2]. Le deuxième meilleur système est celui de (ElSahar & El-Beltagy, 2015). Les bonnes performances de ce système s'expliquent notamment par l'utilisation de connaissances de type expert *a priori* relatives à la polarité par le moyen de lexiques, malheureusement non disponibles. Nous comparons donc les résultats de nos systèmes à l'exactitude de (Dahou *et al.*, 2016) sur corpus officiel, qui correspond au meilleur résultat obtenu sans connaissances *a priori* (**soit une baseline à 77,39%**).

Notre premier système s'appuie sur une implémentation de CNN similaire à celle de (Dahou *et al.*, 2016). En plus des *embeddings* de (Dahou *et al.*, 2016), nous avons également testé les *embeddings* de (Soliman *et al.*, 2017) décrits dans la section 3.2. Notre second système s'appuie sur un LSTM et a été testé avec les différents *embeddings*.
Les performances de ces différentes combinaisons architecture/*embeddings* sont résumées dans la table 1. Elles nous permettent d'étudier de façon exploratoire l'impact de différentes constructions de représentations continues de mots sur la détection d'opinion. Nous notons ici que le CNN obtient de meilleurs résultats que le LSTM, et ce, quels que soient les *embeddings* utilisés. La meilleure performance est atteinte par un CNN appris sur les *embeddings* de (Soliman *et al.*, 2017) avec une approche CBOW appliquée sur un corpus issu du Web. Ce système noté *CNN_Soliman_CBOW_Web* sera analysé dans la section suivante.

2. En utilisant leurs partitions personnelles du corpus du LABR, nous retrouvons leurs résultats.

	(Dahou *et al.*, 2016)	(Soliman *et al.*, 2017)					
	Web	**Twitter**		**Wikipédia**		**Web**	
	CBOW	CBOW	Skip-gram	CBOW	Skip-gram	CBOW	Skip-gram
CNN	77,39%	77,41%	77,55%	77,51%	77,43%	77,56%	77,47%
LSTM	75,03%	74,87%	74,65%	74,92%	74,58%	74,74%	74,95%

TABLE 1 – Exactitudes des architectures CNN et LSTM sur LABR avec différents *embeddings*.

On remarque également que les *embeddings* obtiennent tous des résultats similaires malgré le fait que certains *embeddings* n'ont pas été appris avec la même approche ou le même type de corpus. Nous nous interrogeons ainsi sur la pertinence des représentations de mots disponibles pour la tâche spécifique de la détection d'opinions. Dans la section suivante, nous analysons dans un premier temps les erreurs de notre meilleur système puis proposons une première analyse des *embeddings* utilisés.

5 Analyse des résultats

5.1 Analyse des erreurs de prédiction

Nous avons calculé la matrice de confusion de notre meilleur système, *CNN_Soliman_CBOW_Web*. Le système prédit bien les commentaires positifs avec 80,34% de précision et 89,80% de rappel. Les exemples négatifs sont, quant à eux, plus difficiles à détecter avec 67,76% de précision et seulement 49,37% de rappel. Notre système montre donc une faiblesse dans la prédiction de la classe négative.

Pour analyser plus finement la composition des critiques, nous nous appuyons sur les mots issus du lexique *LABR_lex* de (ElSahar & El-Beltagy, 2015) qui regroupe 873 expressions [3] dont la polarité est connue. On dit que ce sont des mots *polarisés*. Les mots de ce lexique constituent 2,4% des occurrences de mots contenus dans les critiques positives ou négatives du corpus LABR. La majorité ($\geq$ 1,6%) de ces mots sont des mots positifs. La difficulté de classification des critiques négatives peut donc être due à l'utilisation de figures de styles comme l'humour ou l'ironie qui implique qu'une expression positive est utilisée alors que le sens se veut négatif. Une autre explication à l'apparition de ces mots positifs dans une critique négative est qu'ils sont utilisés en conjonction avec un terme de négation. Nous avons par exemple remarqué que parmi les vingt mots les plus fréquents, trois étaient des termes de négation. Nous pensons également que la difficulté de classification des critiques négatives peut être fortement liée à la pertinence des *embeddings* d'entrée pour la tâche donnée. Nous proposons dans la section suivante un protocole d'analyse afin d'étudier cette hypothèse.

5.2 Analyse des *embeddings*

Dans un premier temps, nous proposons de calculer la couverture des mots du corpus LABR par les projections existantes dans l'un des 7 espaces d'*embeddings* considérés. Pour ce faire, nous avons considéré d'une part tous les mots puis d'autre part les mots les plus fréquents (nombre d'occurrences >5), et calculé les couvertures d'une part sur le vocabulaire du corpus LABR (Table 3) et sur le corpus lui-même (Table 2).

Nous remarquons que la couverture du corpus par les différents espaces d'*embeddings* se situe aux alentours de 60% quels que soient l'espace considéré. La couverture augmente de six à huit points si on ne considère que les mots fréquents. Au niveau du vocabulaire, plus de 55% des mots fréquents

3. Une expression dans le lexique peut être constituée d'un ou plusieurs mots.

corpus LABR	(Dahou *et al.*, 2016)	(Soliman *et al.*, 2017)					
	Web	**Twitter**		**Wikipédia**		**Web**	
	CBOW	CBOW	Skip-Gram	CBOW	Skip-Gram	CBOW	Skip-Gram
tous	67,33%	60.27%	60.87%	61.53%	61.53%	60.35%	60.19%
occur > 5	71,07%	66.04%	66.32%	68.06%	68.06%	66.23%	66.07%

TABLE 2 – Couverture du corpus LABR par les différents modèles d'embeddings.

vocabulaire LABR	(Dahou *et al.*, 2016)	(Soliman *et al.*, 2017)					
	Web	**Twitter**		**Wikipédia**		**Web**	
	CBOW	CBOW	Skip-Gram	CBOW	Skip-Gram	CBOW	Skip-Gram
tous	40,97%	22.11%	24.33%	19.16%	19.16%	21.45%	21.30%
occur > 5	64,89%	57.00%	58.44%	53.48%	53.48%	57.76%	57.38%

TABLE 3 – Couverture du vocabulaire de LABR par les différents modèles d'embeddings.

sont couverts alors que la couverture du vocabulaire chute à 20% si on considère tous les mots. Ceci indique que la grande majorité des mots du corpus LABR n'ayant pas d'*embeddings* dans les modèles disponibles sont des mots peu fréquents. Ainsi, bien que la couverture ne soit pas très grande elle semble suffisante pour la classification.

Dans un second temps, afin d'évaluer la pertinence dans le cadre spécifique de la tâche d'analyse d'opinions des représentations de mots dans un espace continu, nous proposons d'étudier la polarité des mots voisins, en considérant leur *embeddings* dans chacun des espaces, pout les mots polarisés. Pour chaque expression, son ensemble des n plus proches mots polarisés voisins (Top$_n$) dans l'espace d'embeddings, est considéré selon la similarité cosinus. Nous calculons alors un ratio de *positivité* des mots de polarisés associés à une polarité positive ($lexique^+$) (voir équation 1).

$$\%^+_{Top_n} = 100 \times \frac{\sum_{mot_i \in \{lexique^+\}} \#mot^{lexique^+}_{i,Top_n}}{n \times \#lexique^+} \quad (1)$$

avec : n le nombre de mots voisins considérés ; $\#mot^{lexique^+}_{i,Top_n}$ le nombre de mots positifs parmi les n plus proches voisins du mot i du corpus $lexique^+$; $\#lexique^+$ le nombre de mots positifs dans *lexique*.

Nous calculons également un ratio de *négativité* selon la même formule en ne considérant que les mots négatifs. Nous considérons qu'une représentation pertinente des mots dans un espace continu pour la tâche de détection d'opinions projetterait les mots positifs dans la même zone et les mots négatifs dans une autre zone. On observerait alors un ratio proche de 100%.

La Table 4 montre les résultats du ratio de positivité calculé sur le lexique *LABR_lex*. Nous constatons que plus le voisinage considéré est large, plus le ratio de positivité est grand. Ceci signifie que les mots positifs sont de plus en plus entourés par des mots positifs du lexique. En revanche, pour le ratio de négativité, calculé également à l'aide du lexique *LABR_lex*, nous constatons que plus le voisinage est large, moins le mot négatif est entouré de mots négatifs. Etant donné que seuls les mots polarisés sont considérés, ceci signifie que les mots négatifs sont de plus en plus entourés par des mots

		(Dahou *et al.*, 2016)	(Soliman *et al.*, 2017)					
		Web	**Twitter**		**Wikipédia**		**Web**	
		CBOW	CBOW	Skip-G	CBOW	Skip-G	CBOW	Skip-G
$\%^{+}_{Top_n}$	n=2	43,13	41,79	41,48	38,83	38,83	42,59	41,15
	n=5	68,10↗	63,28 ↗	64,88 ↗	58,75 ↗	58,21 ↗	66,66 ↗	63,23 ↗
	n=10	73,46↗	68,43 ↗	69,92 ↗	63,39 ↗	62,14 ↗	72,07 ↗	70 ↗
$\#LABR\_lex^{+}$		153	134	135	112	112	135	130
$\%^{-}_{Top_n}$	n=2	34,88	38,75	42,42	31,77%	37,85%	39,09%	40,83%
	n=5	13,95↘	15,5↘	16,96↘	12,71↘	15,14↘	15,63↘	16,33↘
	n=10	6,97↘	7,75↘	8,84↘	6,35↘	7,57↘	7,81↘	8,16↘
$\#LABR\_lex^{-}$		172	160	165	107	107	133	131

TABLE 4 – Ratios de *positivité* (respectivement *négativité*) des mots positifs (respectivement négatifs) dont l'embedding existe à la fois dans *LABR_lex* et le corpus d'*embeddings* considéré.

positifs du lexique. Ces observations se vérifient pour les différents espaces d'*embeddings*. La polarité négative semble donc diffusée dans l'espace de représentations utilisé. Ceci appuie notre hypothèse d'un espace continu non adapté au cadre de la détection d'opinions, notamment pour représenter les mots négatifs. Cette observation explique les mauvais résultats en classification d'opinions des commentaires négatifs.

6 Conclusion et perspectives

Dans cet article, nous avons étudié l'utilisation de techniques d'apprentissage profond dans le cadre de l'analyse d'opinion pour l'arabe en étudiant sept ensembles d'*embbeddings* différents comme entrées de réseaux CNN et LSTM.

Nos expériences ont montré que l'architecture CNN est plus performante que l'architecture LSTM, quelque soit le modèle d'*embeddings* utilisé. Notre meilleur système ($CNN\_Soliman\_CBOW\_Web$) obtient une exactitude de 77,56% améliorant légèrement le meilleur système publié qui n'utilise pas de connaissances *a priori* (77,39% pour (Dahou *et al.*, 2016) appliqué sur la répartition officielle).

Nous proposons trois pistes d'amélioration de ces premiers travaux : (i) utilisation de formes fléchies des mots. En effet, plus de 80% des mots sont peu fréquents, une lemmatisation permettrait d'éviter la dispersion du vocabulaire ; (ii) création automatique de lexiques de mots polarisés. Les meilleurs résultats ont été obtenus avec des connaissances *a priori* coûteuses à obtenir. Nous souhaitons étudier la traduction de ressources existantes afin de créer un ensemble de connaissances *a priori* pour l'arabe ; (iii) création d'*embeddings* spécifiques. Notre analyse des *embeddings* génériques disponibles a montré que ceux-ci n'étaient pas forcément pertinents pour notre tâche. En nous appuyant sur les travaux de (Yu *et al.*, 2017) où des *embeddings d'opinions* sont construits pour l'anglais, nous souhaitons étudier la transposition de ces travaux pour l'arabe en nous appuyant sur les lexiques de mots polarisés que nous aurons construits.

Références

ABDULLA N. A., AHMED N. A., SHEHAB M. A., AL-AYYOUB M., AL-KABI M. N. & AL-RIFAI S. (2014a). Towards improving the lexicon-based approach for arabic sentiment analysis. *International Journal of Information Technology and Web Engineering (IJITWE)*, **9**(3), 55–71.

ABDULLA N. A., AL-AYYOUB M. & AL-KABI M. N. (2014b). An extended analytical study of arabic sentiments. *International Journal of Big Data Intelligence 1*, **1**(1-2), 103–113.

AL-KABI M., AL-AYYOUB M., ALSMADI I. & WAHSHEH H. (2016). A prototype for a standard arabic sentiment analysis corpus. *Int. Arab J. Inf. Technol.*, **13**(1A), 163–170.

AL-KABI M. N., GIGIEH A. H., ALSMADI I. M., WAHSHEH H. A. & HAIDAR M. M. (2014). Opinion mining and analysis for arabic language. *International Journal of Advanced Computer Science and Applications (IJACSA), SAI Publisher*, **5**(5), 181–195.

ALMAS Y. & AHMAD K. (2007). A note on extracting 'sentiments' in financial news in english, arabic & urdu. In *The Second Workshop on Computational Approaches to Arabic Script-based Languages*, p. 1–12.

BARHOUMI A., ESTÈVE Y., ALOULOU C. & BELGUITH L. H. (2017). Document embeddings for arabic sentiment analysis. In *Proceedings of the First Conference on Language Processing and Knowledge Management, LPKM 2017, Kerkennah (Sfax), Tunisia, September 8-10, 2017*.

BAYOUDHI A., GHORBEL H. & BELGUITH L. H. (2015). Sentiment classification of arabic documents : Experiments with multi-type features and ensemble algorithms. In *Proceedings of the 29th Pacific Asia Conference on Language, Information and Computation*, p. 196–205.

DAHOU A., XIONG S., ZHOU J., HADDOUD M. H. & DUAN P. (2016). Word embeddings and convolutional neural network for arabic sentiment classification. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics : Technical Papers*, p. 2418–2427.

DERIU J., LUCCHI A., DE LUCA V., SEVERYN A., MÜLLER S., CIELIEBAK M., HOFMANN T. & JAGGI M. (2017). Leveraging large amounts of weakly supervised data for multi-language sentiment classification. *International World Wide Web Conference Committee (IW3C2)*.

EL-HALEES A. (2011). Arabic opinion mining using combined. *Proceeding the International Arab Conference On Information Technology*.

ELSAHAR H. & EL-BELTAGY S. R. (2015). Building large arabic multi-domain resources for sentiment analysis. In *International Conference on Intelligent Text Processing and Computational Linguistics*, p. 23–34 : Springer.

FARRA N., CHALLITA E., ASSI R. A. & HAJJ H. (2010). Sentence-level and document-level sentiment mining for arabic texts. In *Data Mining Workshops (ICDMW), 2010 IEEE International Conference on*, p. 1114–1119 : IEEE.

GAURANGI P., VARSHA G., VEDANT K. & KALPANA D. (2014). Sentiment analysis using support vector machine. *International Journal of Innovative Research in Computer and Communication Engineering*.

HASSAN A. (2017). Sentiment analysis with recurrent neural network and unsupervised neural language model.

IBRAHIM H. S., ABDOU S. M. & GHEITH M. (2015). Sentiment analysis for modern standard arabic and colloquial. *International Journal on Natural Language Computing (IJNLC)*, **4**(2).

KIM Y. (2014). Convolutional neural networks for sentence classification. *arXiv preprint arXiv :1408.5882*.

MIKOLOV T., SUTSKEVER I., CHEN K., CORRADO G. S. & DEAN J. (2013). Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, p. 3111–3119.

NABIL M., ALY M. & ATIYA A. (2014). Labr : A large scale arabic sentiment analysis benchmark. *arXiv preprint arXiv :1411.6718*.

PANG B., LEE L. *et al.* (2008). Opinion mining and sentiment analysis. *Foundations and Trends® in Information Retrieval*, **2**(1–2), 1–135.

REFAEE E. & RIESER V. (2016). ilab-edinburgh at semeval-2016 task 7 : A hybrid approach for determining sentiment intensity of arabic twitter phrases. *Proceedings of SemEval-2016*, p. 474–480.

SOLIMAN A. B., EISSA K. & EL-BELTAGY S. R. (2017). Aravec : A set of arabic word embedding models for use in arabic nlp. *Procedia Computer Science*, **117**, 256–265.

TRIPATHY A., AGRAWAL A. & RATH S. K. (2015). Classification of sentimental reviews using machine learning techniques. *3rd International Conference on Recent Trends in Computing (ICRTC-2015)*, p. 821—829.

WILSON T., WIEBE J. & RWA R. (2004). Just how mad are you ? finding strong and weak opinion clauses. *Proceedings of the Nineteenth National Conference on Artificial Intelligence, Sixteenth Conference on Innovative Applications of Artificial Intelligence*, p. 761–769.

YU L.-C., WANG J., LAI K. R. & ZHANG X. (2017). Refining word embeddings for sentiment analysis. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, p. 534–539.

ZAINUDDIN N. & SELAMAT A. (2014). Sentiment analysis using support vector machine. *International Conference on Computer, Communications, and Control Technology (I4CT)*.

ZHOU P., QI Z., ZHENG S., XU J., BAO H. & XU B. (2016). Text classification improved by integrating bidirectional lstm with two-dimensional max pooling. *arXiv preprint arXiv :1611.06639*.

Évaluation automatique de la satisfaction client à partir de conversations de type "chat" par réseaux de neurones récurrents avec mécanisme d'attention

Jeremy Auguste[1] Delphine Charlet[2] Géraldine Damnati[2] Benoit Favre[1]
Frédéric Béchet[1]
(1) Aix-Marseille Univ, Université de Toulon, CNRS, LIS, Marseille, France
(2) Orange Labs, Lannion
(1) {jeremy.auguste,benoit.favre,frederic.bechet}@lis-lab.fr
(2) {delphine.charlet,geraldine.damnati}@orange.fr

RÉSUMÉ

Cet article présente des méthodes permettant l'évaluation de la satisfaction client à partir de très vastes corpus de conversation de type "chat" entre des clients et des opérateurs. Extraire des connaissances dans ce contexte demeure un défi pour les méthodes de traitement automatique des langues de par la dimension interactive et les propriétés de ce nouveau type de langage à l'intersection du langage écrit et parlé. Nous présentons une étude utilisant des réponses à des sondages utilisateurs comme supervision faible permettant de prédire la satisfaction des usagers d'un service en ligne d'assistance technique et commerciale.

ABSTRACT

Customer satisfaction prediction with attention-based RNNs from a chat contact center corpus

This paper presents methods to perform knowledge extraction from very large databases of WEB chat conversations between operators and clients in customer contact centers. Extracting knowledge from chat corpus is a challenging research issue. Simply applying traditional text mining tools is clearly sub-optimal as it takes into account neither the interaction dimension nor the particular nature of this language which shares properties of both spoken and written language. We present a method predicting users satisfaction in a chat-based service trained on answers from users to satisfaction surveys.

MOTS-CLÉS : Réseaux de neurones récurrents, Attention, Satisfaction client, Conversations.

KEYWORDS: Recurrent Neural Networks, Attention-based RNNs, chat, satisfaction prediction.

1 Introduction

L'analyse automatique d'enregistrements de conversations, écrites ou orales, représente un défi pour les méthodes de Traitement Automatique de la Langue à cause d'une part de la nature spontanée du langage employé, et d'autre part de la construction interactive du discours au fur et à mesure de l'échange entre les participants. En dehors des tâches d'extraction d'informations telles que la classification de conversations (Koço *et al.*, 2012) ou le résumé automatique (Trione *et al.*, 2016),

l'une des tâches les plus étudiées dans le cadre des conversations *avec but* est celle de l'évaluation de la *satisfaction* des intervenants, en particulier dans le contexte applicatif des centres de contact clientèle des entreprises ou des administrations.

En effet les services techniques, commerciaux ou juridiques de la majorité des grands groupes peuvent être contactés par l'intermédiaire de centres d'appels ou grâce à des systèmes de conversation textuels hébergés sur leurs sites web. Ce dernier type de conversations, dites conversations *médiées*, sont faciles à collecter et une étude poussée de leur contenu permet de contrôler et d'améliorer la qualité des services rendus. Le champ de l'*analytics* s'intéresse à ces aspects avec par exemple l'étude manuelle par des experts d'échantillons de conversations, l'extraction de statistiques à partir de l'analyse automatique du contenu. Cependant une grande partie des informations sont le plus souvent issues de l'analyse des sondages de satisfaction, sous forme de questionnaires soumis aux usagers suite aux conversations. La réponse aux questions de type "*Votre problème a-t-il été résolu ?*" ou "*Recommanderiez-vous le service à vos proches ?*" sont des indicateurs de performance importants pour les services concernés. Une des limites de ce type d'étude est dû au fait que de nombreux usagers ne prennent pas la peine de répondre aux questionnaires de satisfaction. Se limiter aux seuls sondages renseignés donne une vue tronquée de la réalité d'un service.

Nous nous intéressons dans cette étude à la problématique de la prédiction automatique des réponses aux enquêtes de satisfaction à partir des seules transcriptions de conversations. Nous décrivons un système de prédiction automatique d'indicateurs de qualité entraîné sur des corpus de questionnaires post-conversations. Cette tâche ouvre plusieurs perspectives applicatives comme la prédiction de la satisfaction pour les clients ne répondant pas aux enquêtes. Elle constitue également un travail préalable à la mise en œuvre de monitoring en temps réel de la satisfaction client.

2 Travaux connexes

L'évaluation de la satisfaction client au sein de centres d'appels a donné lieu à de nombreuses études, la plupart sur des conversations orales téléphoniques. Les critères utilisés pour mesurer cette satisfaction peuvent être multiples, de critères objectifs tels que la réalisation effective de la tâche ayant motivé l'appel ou le temps d'attente, jusqu'aux critères subjectifs relatifs à la perception de l'efficacité ou la capacité d'écoute du téléconseiller ayant géré l'appel, ou encore la volonté de l'utilisateur de recommander ou pas le service qu'il vient d'utiliser, ce dernier critère étant l'un des plus importants pour les entreprises concernées (Reichheld, 2003).

Les systèmes qui ont été développés pour prédire automatiquement ces critères peuvent utiliser deux types de supervisions pour entraîner leurs modèles : une supervision directe sous la forme de sondages demandant aux utilisateurs de répondre à une enquête de satisfaction immédiatement après une conversation ; une supervision indirecte en demandant à des experts d'évaluer la satisfaction des appelants perçue à partir des transcriptions. C'est généralement la supervision indirecte qui est utilisée en raison des difficultés à obtenir les enquêtes de satisfaction des clients. Par exemple le système QA^{RT} décrit dans (Roy *et al.*, 2016) permet de prédire directement la qualité d'une conversation, au fur et à mesure de son déroulement, à l'aide d'indices et de classifieurs entraînés sur des avis d'experts. Cette prédiction de satisfaction à base d'avis d'experts a aussi été utilisée pour évaluer le ressenti d'utilisateurs d'interfaces de dialogue homme-machine, comme récemment dans (Stoyanchev *et al.*, 2017) où des classifieurs à base de Support Vector Machine (SVM) sont utilisés, ou encore dans (Pragst *et al.*, 2017) où des approches à base de réseaux de neurones récurrents (RNN) permettent de

prendre en compte la séquentialité des tours de parole dans une conversation.

L'utilisation d'une supervision indirecte, sous forme d'avis d'experts, pose problème pour évaluer des critères aussi subjectifs que la perception d'une qualité d'écoute, ou la volonté de recommander un service. Ce point est discuté dans (Ultes *et al.*, 2013) où il est montré qu'il existait une bonne corrélation entre les enquêtes d'opinion et ces avis d'experts. Cependant l'une des principales limitations des études précédentes est la faible taille des corpus utilisés. En effet, même pour les avis d'experts, obtenir des annotations sur de grands corpus reste une tâche difficile et coûteuse.

L'une des principales originalités de notre étude est d'avoir pu utiliser de très grands corpus de dialogue avec une *supervision directe* sous la forme de sondages de satisfaction effectués à l'issue des conversations. En effet le format des conversations "*chat*", ainsi que le volume des conversations disponibles font que nous pouvons disposer d'un très grand ensemble de sondages, sur lesquels des modèles de prédiction sont entraînés. Dans (Hara *et al.*, 2010), les auteurs exploitent des enquêtes de satisfaction renseignées par les utilisateurs eux-même afin d'estimer la satisfaction face à un système de dialogue homme-machine. Cependant, ces questionnaires étaient dans leur cas directement liés à la qualité perçue du système de dialogue, avec des questions orientées dans ce sens. Ici, nous disposons d'un volume important de conversations "*chat*" avec une annotation directe de satisfaction selon plusieurs dimensions.

Disposer d'une supervision directe est bien évidemment un atout majeur, cependant cela pose des questions sur la difficulté de la tâche de prédiction : en effet, contrairement à la supervision indirecte effectuée par des experts se basant uniquement sur des transcriptions de conversation, nous ne savons pas dans quelle mesure les notes données par les utilisateurs ont des traces *objectives* dans la conversation elle-même ou bien proviennent d'un ressenti et d'une expérience utilisateur prenant en compte l'historique des rapports entre le client et le service. Cette étude se propose d'essayer de prédire cette supervision directe de la satisfaction des utilisateurs d'un service, en présentant tout d'abord dans le paragraphe suivant le type de données et de sondage auxquels nous avons eu accès.

3 Conversations et métadonnées

Les conversations utilisées sont issues des logs de conversations de type "chats" provenant du service client de l'entreprise Orange. Les différentes conversations portent sur plusieurs sujets, à la fois techniques sur les problèmes rencontrés avec les services proposés, ou encore des questions à propos d'une offre commerciale. Ces conversations textuelles étant directement issues des "*chat*" entre clients et téléconseillers, il est important de noter qu'il y a une présence assez importante de fautes d'orthographe et autres types de *bruits*. Lorsqu'une conversation avec un agent est terminée, le client a la possibilité de remplir un questionnaire contenant les 5 questions suivantes :

Question	Alias
J'ai été accompagné(e) et j'ai eu les explications pour faire par moi-même	Accompagnement
J'ai été écouté(e) et ma demande a été prise en charge	Ecoute
J'ai été bien conseillé(e)	Conseil
La solution proposée par Orange me convient	Solution
Suite à votre contact avec le Service Clients, recommanderiez-vous Orange à vos proches ?	Recommander

Si certaines questions portent directement sur l'interaction en elle même ("*Accompagnement*",

"*Ecoute*", "*Conseil*"), d'autres ne sont qu'indirectement liées. Ainsi "*Solution*" peut être liée à l'expérience du client à l'issue de la conversation. Enfin, la question "*Recommander*" relève également d'une appréciation générale pour lesquels les clients peuvent exprimer un ressenti plus large que celui qui résulte de la simple conversation. Pour cette question "*Recommander*", le client doit répondre sur une échelle allant de 0 à 10. Suivant les conventions du domaine de l'analyse de la relation client, nous avons réalisé des regroupements pour définir trois catégories : *détracteurs* (note de 0 à 6), *passifs* (7 ou 8) et *promoteurs* (9 ou 10). Pour les autres questions, le client doit répondre sur une échelle à 5 niveaux allant de "Pas du tout satisfait(e)" à "Très satisfait(e)". Les réponses à ces questions sont des indicateurs importants pour juger de la qualité de service.

Les données ont été collectées sur une période d'un mois et nous avons sélectionné le sous-ensemble de conversations pour lesquelles nous avions une réponse à toutes les questions. Les corpus d'entraînement, de développement et de test sont respectivement constitués de 47685, 15899 et 15892 conversations. Le corpus d'entraînement est composé de 140000 tokens différents. Comme précisé dans le paragraphe précédent, l'originalité de l'étude est que la supervision de l'annotation est faite directement par le client. Il y a ainsi autant d'annotateurs que de conversations. La quantité très importante de données d'apprentissage et de test disponibles avec cette supervision directe (près de 80K conversations), est aussi très inhabituelle pour ce type d'étude ou ce sont généralement de petits volumes qui sont considérés.

4 Prédiction automatique de la satisfaction client

L'objectif est d'évaluer dans quelle mesure il est possible, à partir de l'analyse du contenu des conversations, de prédire automatiquement les réponses aux 5 questions posées à l'issue de ces conversations. Dans une première approche, nous considérons ce problème comme une tâche de classification où pour chaque dimension considérée, un classifieur doit prédire la réponse à la question posée. Les classifieurs diffèrent entre autres par le mode de représentation du texte des conversations traitées : simples sacs de mots (1) contenant toute la conversation, découpage en blocs contigus (2) ou séquences de mots ordonnées (3).

Pour évaluer le premier mode en sac de mots (1), nous utilisons un classifieur SVM avec un modèle pour chaque tâche, dans l'implémentation des SVM à noyau linéaire de Pedregosa *et al.* (2011).

Pour la représentation en bloc (2), nous utilisons un réseau de neurones convolutionnels (CNN) basé sur le réseau décrit par Kim (2014). Nous créons un modèle par tâche ayant des filtres de tailles 3, 4 et 5 avec 100 filtres pour chaque taille.

Pour le dernier mode prenant en compte l'ordre des mots (3), nous avons implémenté un réseau de neurones récurrent (RNN) de type *Long Short Term Memory* (LSTM). Etant donné la forte variabilité inhérente à ce mode de représentation où chaque conversation est représentée comme une unique séquence de tokens (environ 500 mots en moyenne par conversation), nous avons ajouté en complément au RNN, un *mécanisme d'attention* (Bahdanau *et al.*, 2014; Xu *et al.*, 2015) permettant au système de se focaliser sur les mots importants d'une conversation par rapport à la tâche visée. Dans cette configuration, les interventions de l'usager et du téléconseiller sont concaténées en ajoutant un token <EOT> à la fin de chaque tour de parole. En sortie, le réseau donne, pour chaque conversation, une distribution de probabilité sur l'ensemble des classes de la tâche. La classe sélectionnée est celle qui a la plus haute probabilité.

Le mécanisme d'attention utilisé est le suivant :

$$u_t = v^\top \tanh(W_a h_t + b_a) \qquad \alpha_t = \frac{\exp(u_t)}{\sum_{i=1}^{n} \exp(u_i)} \qquad \text{SelfAttn}(h) = \sum_{i=1}^{n} \alpha_i h_i$$

où W_a et b_a sont des paramètres de la fonction calculant le score d'attention et v est le vecteur de contexte qui est aléatoirement initialisé et qui est également entraîné lors de la phase d'apprentissage.

Soit $W = w_1 \dots w_n$ une conversation. Les distributions de probabilité p sont obtenus avec :

$$x_t = \text{Embedding}(w_t) \quad (1) \qquad \overrightarrow{h}_t = \overrightarrow{\text{LSTM}}(x_t, \overrightarrow{h}_{t-1}) \quad (2) \qquad \overleftarrow{h}_t = \overleftarrow{\text{LSTM}}(x_t, \overleftarrow{h}_{t-1}) \quad (3)$$

$$h_t = [\overrightarrow{h}_t, \overleftarrow{h}_t] \quad (4) \qquad h = \{h_t \mid t \in [1,n]\} \quad (5) \qquad c = \text{SelfAttn}(h) \quad (6) \qquad p = \text{softmax}(W_d c + b_d) \quad (7)$$

Dans les équations précédentes, LSTM est une couche de Long Short-Term Memory units (Hochreiter & Schmidhuber, 1997). Le sens de la flèche indique le sens de lecture des séquences par la couche LSTM. W_d et b_d sont les paramètres de la couche de décision.

Dans le réseau implémenté, les couches cachées des LSTMs sont de taille 128. Les *embeddings* de mots sont de dimensions 100. L'information indiquant qui est le scripteur du tour de parole (client, téléconseiller, système) est également intégrée sous forme d'*embeddings* de taille 3 concaténés aux *embeddings* de mots. Nous utilisons la fonction d'entropie croisée pour la fonction de coût. Les poids du modèle sont initialisés uniformément dans $[-0.1,0.1]$, et optimisés avec l'algorithme ADAM. Un dropout de 0.5 est appliqué après la couche de LSTMs. Pour des contraintes techniques, les tailles des conversations sont normalisées à une taille de 1200 mots. Pour les conversations plus courtes, un symbole de *padding* est utilisé pour compléter les conversations ; pour celles qui sont plus longues, les 1200 derniers tokens sont pris en compte. Dans l'ensemble du corpus, seules 4% des conversations sont partiellement coupées.

5 Expérimentations

La prédiction de la satisfaction est abordée dans cette étude comme un problème de classification, on mesure donc la performance de la prédiction par le taux de labels correctement prédits, appelée *accuracy* dans la suite de l'étude. Dans ce type d'évaluation, il n'est pas plus grave de faire une confusion entre le label "0" et le label "4" qu'entre le label "0" et le label "1". Cependant, d'un point de vue applicatif, ces confusions n'ont pas la même valeur. Les labels étant des notes sur une échelle de satisfaction graduée, il est plus grave de considérer un client pas du tout satisfait comme très satisfait, plutôt que comme peu satisfait.

C'est pourquoi nous évaluons également les classifieurs avec des mesures qui exploitent la gradation des labels, qui sont ici des valeurs numériques ordonnées (que l'on peut qualifier de notes). Nous utilisons pour cela le coefficient de corrélation de Spearman entre l'ensemble des notes prédites et l'ensemble des notes réelles car il permet d'évaluer si les notes prédites conservent l'ordonnancement des notes réelles. En effet, ce coefficient de corrélation vaut 1 quand il existe une fonction monotone croissante entre labels prédits et labels réels, c'est-à-dire si les ordonnancements des labels prédits sont identiques aux ordonnancements des labels réels. Afin d'avoir une mesure d'évaluation plus interprétable, nous utilisons également la mesure Δ_{abs}, définie comme la moyenne de la valeur absolue de la différence entre le label réel (score de satisfaction de la vérité-terrain) et le label prédit. Cette mesure doit être la moins élevée possible, un classifieur parfait rendant une mesure de 0.

satisfaction (taille de l'échelle)	approche	accuracy	spearman	Δ_{abs}
Accompagnement (5)	Majorité	48.48	-	0.974
	SVM	55.28	0.542	0.729
	CNN	**56.85**	0.549	0.64
	RNN	55.68	0.553	0.644
	RNN+Attn	56.82	**0.57**	**0.633**
Conseil (5)	Majorité	53.24	-	0.867
	SVM	59.84	0.517	0.613
	CNN	61.22	0.568	0.562
	RNN	60.56	0.565	0.569
	RNN+Attn	**61.43**	**0.594**	**0.556**
Solution (5)	Majorité	44.38	-	1.195
	SVM	54.62	0.544	0.788
	CNN	55.82	0.587	0.724
	RNN	54.12	0.574	0.767
	RNN+Attn	**56.21**	**0.611**	**0.713**
Ecoute (5)	Majorité	54.57	-	0.833
	SVM	61.26	0.517	0.613
	CNN	62.77	0.554	0.54
	RNN	61.91	0.556	0.554
	RNN+Attn	**63.10**	**0.570**	**0.532**
Recommander (3)	Majorité	42.71	-	0.882
	SVM	56.31	0.441	0.593
	CNN	56.27	0.468	**0.562**
	RNN	56.01	0.435	0.605
	RNN+Attn	**57.53**	**0.478**	0.581

TABLE 1 – Résultats des classifieurs pour la prédiction des indicateurs de qualité. Accuracy et Spearman doivent être les plus élevés possibles et Δ_{abs} doit être idéalement proche de 0.

Le tableau 1 présente les résultats des différents classifieurs, pour prédire les réponses aux différentes questions sur la satisfaction. On reporte également les résultats obtenus en attribuant simplement la classe majoritaire à tous les exemples (approche *Majorité* dans le tableau). Cette *baseline* permet de vérifier que le déséquilibre entre les classes n'est pas trop important (au maximum une classe couvre environ 50% des exemples).

On peut constater dans la table 1 qu'en utilisant un RNN simple sans attention, on obtient des scores d'accuracy équivalents à $\pm 0{,}5$ points près, aux scores obtenus à l'aide du SVM. Cependant, lorsque les mécanismes d'attention sont utilisés, on observe des gains permettant de gagner entre $1{,}2$ et $1{,}8$ points par rapport au SVM. Ces meilleurs résultats indiquent que la présence ou non de certains mots dans une conversation sont de forts indicateurs pour la prédiction de la satisfaction. Au contraire, l'ordre des mots de la conversation l'est moins comme on peut le voir avec les résultats obtenus par les réseaux récurrents sans attention. En comparant les résultats obtenus par le CNN avec ceux obtenus par le RNN avec attention, nous pouvons constater que le CNN obtient des scores inférieurs ou égaux obtenant jusqu'à $1{,}26$ points de moins sur la tâche "**Recommander**".

En regardant de plus près les mots qui obtiennent le plus souvent le plus haut score d'attention, on remarque que les mots en lien avec les remerciements reviennent le plus souvent pour la majorité des tâches. Pour la tâche "**Ecoute**", on constate également qu'il y a plusieurs mots portant sur l'agent comme "*efficacité*" ou "*gentillesse*". Pour la tâche "**Solution**", on observe la présence des mots "*aide*", "*satisfait*" et "*navrée*" faisant probablement référence à la résolution ou non du problème.

Sur les deux autres mesures, on peut observer que le RNN simple obtient de meilleurs performances

que le SVM sur toutes les tâches en excluant la tâche "**Recommander**". On observe que le réseau avec attention obtient les meilleures scores de corrélations sur toutes les tâches avec des gains allant de $0{,}01$ à $0{,}05$ points, ainsi que sur la mesure Δ_{abs}. Dans le cas du CNN, on obtient des scores de corrélations inférieurs aux scores obtenus par le RNN avec attention avec des différences allant de $0{,}01$ à $0{,}02$ points. Ceci est également le cas sur la mesure Δ_{abs} sauf pour la tâche "**Recommander**" où le CNN est meilleur de $0{,}02$ points.

Les tâches obtenant les meilleurs résultats sont "**Conseil**" et "**Ecoute**" ce que l'on pouvait escompter dans la mesure où ce sont les questions les plus directement liées au déroulement de la conversation. "**Accompagnement**" relève d'une appréciation plus subjective et "**Solution**" relève d'une appréciation technique. Quant à la question de savoir si le client recommanderait l'entreprise, il peut y avoir des facteurs subjectifs dépassant le cadre de la simple conversation. Si l'on s'intéresse à l'ordre relatif des jugements, la prédiction de "**Solution**" présente le meilleur coefficient de Spearman, en revanche c'est bien la dimension "**Ecoute**" qui produit les prédictions les plus proches des prédictions réelles en valeur absolue ($\Delta_{abs} = 0.532$).

6 Conclusion

L'évaluation automatique de la satisfaction client à partir d'une conversation n'est pas une tâche facile. Les modèles de type SVMs permettent d'obtenir des résultats raisonnables et les réseaux de neurones convolutionels permettent d'améliorer ces résultats. Cependant, un réseau de neurones récurrents avec un mécanisme d'attention parvient à obtenir de meilleurs résultats que ce soit du point de vue de la classification que de la corrélation de Spearman.

Pour la suite, il serait intéressant de réaliser directement une régression avec le réseau de neurones afin de mieux prendre en compte le fait que les réponses sont données sur une échelle graduée. Il serait aussi intéressant d'essayer d'utiliser des descripteurs structurels en complément des mots.

Remerciements

Ce travail a été partiellement financé par l'Agence Nationale pour la Recherche au sein du projet ANR-15-CE23-0003 (DATCHA).

Références

BAHDANAU D., CHO K. & BENGIO Y. (2014). Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv :1409.0473*.

HARA S., KITAOKA N. & TAKEDA K. (2010). Estimation method of user satisfaction using n-gram-based dialog history model for spoken dialog system. In *LREC*.

HOCHREITER S. & SCHMIDHUBER J. (1997). Long short-term memory. *Neural computation*, **9**(8), 1735–1780. 04135.

KIM Y. (2014). Convolutional neural networks for sentence classification. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, p. 1746–1751.

KOÇO S., CAPPONI C. & BÉCHET F. (2012). Applying multiview learning algorithms to human-human conversation classification. In *Thirteenth Annual Conference of the International Speech Communication Association*.

PEDREGOSA F., VAROQUAUX G., GRAMFORT A., MICHEL V., THIRION B., GRISEL O., BLONDEL M., PRETTENHOFER P., WEISS R., DUBOURG V., VANDERPLAS J., PASSOS A., COURNAPEAU D., BRUCHER M., PERROT M. & DUCHESNAY E. (2011). Scikit-learn : Machine learning in Python. *Journal of Machine Learning Research*, **12**, 2825–2830.

PRAGST L., ULTES S. & MINKER W. (2017). Recurrent neural network interaction quality estimation. In *Dialogues with Social Robots*, p. 381–393. Springer.

REICHHELD F. F. (2003). The one number you need to grow. *Harvard business review*, **81**(12), 46–55.

ROY S., MARIAPPAN R., DANDAPAT S., SRIVASTAVA S., GALHOTRA S. & PEDDAMUTHU B. (2016). Qa rt : A system for real-time holistic quality assurance for contact center dialogues. In *Thirtieth AAAI Conference on Artificial Intelligence*.

STOYANCHEV S., MAITI S. & BANGALORE S. (2017). Predicting interaction quality in customer service dialogs. In *Proceedings of the 2017 INTERNATIONAL WORKSHOP ON SPOKEN DIALOGUE SYSTEMS TECHNOLOGY (IWSD)*.

TRIONE J., FAVRE B. & BÉCHET F. (2016). Beyond utterance extraction : Summary recombination for speech summarization. In *Interspeech*, p. 680–684.

ULTES S., SCHMITT A. & MINKER W. (2013). On quality ratings for spoken dialogue systems–experts vs. users. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics : Human Language Technologies*, p. 569–578.

XU K., BA J., KIROS R., CHO K., COURVILLE A., SALAKHUDINOV R., ZEMEL R. & BENGIO Y. (2015). Show, attend and tell : Neural image caption generation with visual attention. In *International Conference on Machine Learning*, p. 2048–2057.

Détection d'erreurs dans des transcriptions OCR de documents historiques par réseaux de neurones récurrents multi-niveau

Thibault Magallon Frederic Bechet Benoit Favre
Aix Marseille Univ, Université de Toulon, CNRS, LIS, Marseille, France
prenom.nom@lis-lab.fr

RÉSUMÉ
Le traitement à posteriori de transcriptions OCR cherche à détecter les erreurs dans les sorties d'OCR pour tenter de les corriger, deux tâches évaluées par la compétition ICDAR-2017 Post-OCR Text Correction. Nous présenterons dans ce papier un système de détection d'erreurs basé sur un modèle à réseaux récurrents combinant une analyse du texte au niveau des mots et des caractères en deux temps. Ce système a été classé second dans trois catégories évaluées parmi 11 candidats lors de la compétition.

ABSTRACT
Combining character level and word level RNNs for post-OCR error detection

Post-OCR processing, consist in detecting errors first, then correcting them when possible. In this context the ICDAR-2017 Competition on Post-OCR Text Correction was organized to compare approaches on these two tasks. This paper presents an OCR error detection system based on a 2-pass RNN model combining character level and word level representations. This system was ranked 2nd on three datasets among 11 participants at the ICDAR-2017 Competition.

MOTS-CLÉS : OCR, detection d'erreurs, réseaux de neurones récurrents.

KEYWORDS: OCR, error detection, recurrent neural networks.

1 Introduction

Les systèmes actuels de Reconnaissance Optique de Caractères (Optical Character Recognition - OCR) obtiennent désormais d'excellentes performances sur des documents imprimés et scannés avec soin. Cependant les documents historiques restent un défi pour les domaines du traitement d'images et du traitement automatique de la langue du fait d'une mauvaise qualité d'impression ainsi que de supports parfois endommagés. De plus, certaines collections de documents numérisés à l'aide de ces outils ne sont que rarement retraitées avec des systèmes à jours, principalement pour des raisons de coûts. Or ces erreurs d'OCR peuvent avoir un impact conséquent sur la recherche de documents dans une bibliothèque numérique (Chiron *et al.*, 2017b). Ainsi, indépendamment de la tâche d'OCR elle-même, le post-traitement des transcriptions automatiques est une tâche permettant à la fois d'évaluer la qualité d'archives numérisées tout en donnant l'occasion de les corriger. C'est dans ce contexte que la compétition *ICDAR-2017 Post-OCR Text Correction* a été organisée, dans le but de comparer différentes approches concernant les deux tâches de détection d'erreurs d'une part et de correction des erreurs détectées d'autre part. En tenant compte des contraintes liées à ces archives,

seule la sortie de texte brute est accessible, aucune autre information n'est fournie, comme l'image, les scores de confiances ou les informations relatives aux polices détectées.

Historiquement, ces systèmes utilisent des modèles de caractères ainsi que des collections de mots pour résoudre certaines ambiguïtés survenant après l'analyse d'image (Bokser, 1992). Mais il arrive que des erreurs subsistent à l'utilisation de telles méthodes, car le vocabulaire ne peut être couvert pour toute la langue d'une part et pour des questions propres au domaine de l'OCR ou l'utilisation de modèles reposant uniquement sur la fréquence des mots peuvent s'avérer insuffisants (Smith, 2011). De nombreuses méthodes ont été proposées pour simultanément détecter et corriger ces erreurs, en utilisant différentes approches tels que les canaux bruités (Kolak & Resnik, 2002; Evershed & Fitch, 2014), l'utilisation d'outils externes tels que des correcteurs orthographiques (Bassil & Alwani, 2012; Schulz & Kuhn, 2017) ou la combinaison de plusieurs systèmes d'OCR afin d'accroître la robustesse de la reconnaissance (Abdulkader & Casey, 2009). De plus, il n'est pas rare dans le cas de la correction d'erreurs à posteriori d'étiqueter les corrections à apporter sur une séquence donnée afin de la remanier par la suite, comme cela peut être fait pour la post-édition automatique de traductions (Libovickỳ *et al.*, 2016; Bérard *et al.*, 2017).

Le présent document décrit l'architecture proposée par notre équipe pour la *tâche de détection d'erreurs* de la compétition ICDAR. Nous avons suivi le même type d'approche ayant été appliquée avec succès aux méthodes de détection d'erreurs dans des sorties de Reconnaissance Automatique de la Parole (Béchet & Favre, 2013), en considérant la tâche de détection d'erreur comme un exercice de classification de séquences. La méthode repose sur un modèle de réseaux de neurones récurrent analysant le texte à l'échelle des caractères ainsi qu'à celle des mots, le tout joint dans un seul modèle multilingue s'exécutant en deux temps. Ce système a été utilisé sur un corpus rassemblé pour l'occasion contenant des documents historiques en anglais et français et a été classé second dans le cadre de la tâche de détection parmi onze participants.

2 La compétition Post-OCR Text Correction

La compétition ICDAR-2017 Post-OCR (Chiron *et al.*, 2017a) a été séparée en deux tâches afin d'évaluer les différentes méthodes proposées par chaque compétiteur pour la détection ainsi que la correction d'erreurs dans des textes produits par un système OCR.

Le corpus fourni dans ce cadre est composé d'extraits de monographies et de périodiques rédigés en français ou en anglais provenant d'archives de la Bibliothèque Nationale Française (BnF) et de la British Library (BL). Ce jeu de données regroupe au total plus de douze millions de caractères dont les textes sont issus d'une période couvrant les quatre siècles derniers. L'ensemble de ces textes provient d'une sous-partie d'un corpus collecté dans le cadre du projet AmeliOCR, mené par le laboratoire L3i (Université de La Rochelle, France) et la Bibliothèque nationale Française. Ces deux tâches de post-traitement sont difficiles, car si les technologies d'OCR ont aujourd'hui acquis une certaine maturité, la qualité des supports source et la structure des documents ainsi que le vocabulaire ancien et varié ajoute une autre difficulté à l'extraction du texte contenu dans les images numérisées, produisant des sorties textuelles erronées nécessitant une correction.

Afin de pouvoir évaluer les détections et corrections apportées, des annotations *Gold-Standard* (GS) accompagnent les sorties de texte brut. Ces dernières ont étés produites manuellement et sont alignées au niveau des caractères.

La distribution des sources dans ce corpus de 12M de caractères en fonction de la langue et du type de document est référencée dans le Tableau 1. On observe ainsi que le taux d'erreur OCR peut varier de 1% à 4%.

English				
corpus	*type*	*période*	*%erreur*	*taille*
BL Euro NP	périodique	1744 - 1894	4%	1.8 M
BL Monog	monographie	1858 - 1891	1%	1.2 M
GT BnF Eng	monographie	1802 - 1911	2%	3.0 M
French				
corpus	*type*	*période*	*%erreur*	*taille*
Europeana NP	périodique	1814 - 1944	4%	1.0 M
IMPACT	monographie	1821 - 1864	1%	0.4 M
GT BnF Fr	mélangé	1686 - 1943	1%	2.0 M
Digit. BnF	mélangé	1654 - 2000	3%	0.2 M
News other	périodique	1897 - 1934	4%	0.6 M
Monog other	monographie	1689 - 1883	3%	1.8 M

TABLE 1: Distribution du corpus ICDAR Post-OCR.

3 Réseau de neurones récurrent au niveau des mots et des caractères pour la détection d'erreurs

La première étape de notre système consiste en une combinaison de deux modèles de réseaux de neurones récurrents (*Recurrent Neural Network*, abrégé *RNN* par la suite) au niveau des caractères pour associer une étiquette (*correct* ou *erroné*) à chaque symbole d'une séquence, ainsi que d'un modèle de langue dont le rôle est de prédire le caractère suivant étant donné les caractères l'entourant. Ces deux éléments sont présentés dans les sous-sections suivantes. L'implémentation de ce réseau a été réalisée à l'aide du toolkit Keras (Chollet *et al.*, 2015).

3.1 Classification binaire à l'échelle des caractères

Ce premier RNN est une implémentation directe de la tâche de détection d'erreurs au niveau des caractères. C'est-à-dire qu'étant donné un certain contexte, il doit prédire si l'apparition d'un symbole est erroné ou non, sans aucune autre information d'entrée que la séquence textuelle.

Concernant l'entrée du modèle, nous l'avons fixée à un vecteur de dimension 64, où pour tout élément de celui-ci est associé un symbole issu du vocabulaire. Chaque chaîne du texte est alors découpé en plusieurs séquences de ce format (si cela est nécessaire). De plus, une représentation vectorielle de cette entrée est utilisé (plongement de mots) à l'aide d'une fenêtre regroupant les quatorze voisins d'un terme cible.

La sortie de cette couche nous permet d'obtenir une représentation vectorielle de notre séquence d'entrée. Elle est par la suite dirigée vers une couche récurrente afin de considérer les erreurs présente dans celle-ci et ainsi effectuer la tâche de classification associée à la détection d'erreurs. Cette couche récurrente implémente un modèle à mémoire de type *Gated Recurrent Units* (GRU) (Cho *et al.*, 2014).

Nous utilisons 64 neurones pour la couche récurrente, soit autant que de symboles qui composent nos vecteurs d'entrée. De plus, nous ajoutons à cette couche des propriétés de bidirectionnalité ainsi que le fait de ne pas réinitialiser l'état caché pour chaque nouveau vecteur d'entrée (modèle *stateful*). Pour finir, une couche dense de deux unités sur laquelle nous appliquons une activation de type *softmax* est utilisée afin de prédire pour chaque symbole les probabilités d'appartenance aux deux classes, *correct* et *incorrect*.

3.2 Le modèle de langue à l'échelle des caractères

Le modèle récurrent de classification binaire décrit précédemment pourrait-être utilisé tel quel pour la tache de prédiction d'erreurs, cependant ce dernier souffre d'un défaut non-négligeable : il ne peut qu'être entraîné sur un corpus de sorties d'OCR accompagné des erreurs annotées. Cette source de données est rare, même dans le contexte de la compétition et sa quantité est donc restreinte. C'est pourquoi, nous avons utilisé des données complémentaires afin d'améliorer notre méthode de classification, en tirant profit d'une grande quantité de texte provenant d'un corpus de textes journalistiques, nous permettant ainsi d'établir et d'apprendre des statistiques propres aux langues sur lesquelles nous souhaitons détecter les erreurs. Néanmoins, nous ne pouvons directement ajouter de telles données au corpus d'entraînement puisqu'aucun symbole de ce dernier n'est annoté pour des erreurs d'OCR. Mais nous pouvons les ajouter en tant que *Modèle de Langue* (ML), à l'échelle des caractères, afin de prédire le symbole suivant étant donné la position d'un caractère dans une séquence et compte tenu du contexte l'entourant. L'idée derrière l'ajout d'un tel mécanisme est de fournir au système de classification binaire une information supplémentaire quant à la régularité d'apparition de certains termes au travers de la représentation du ML. En effet, un enchaînement peu probable dans la séquence peut modifier le degré de confiance du modèle s'il ne possède qu'une faible probabilité d'apparition compte tenu de sa place dans la séquence et des schémas observés lors de l'apprentissage.

L'entrée du modèle de langue se fait de façon similaire à celle utilisée pour la classification binaire que nous avons décrite précédemment. Concernant sa structure, le réseaux est en tout constitué de trois couches. Une de plongement de mots, une récurrente et une dense d'activation, possédant un nombre d'unités égale au nombre d'éléments présent dans le vocabulaire des symboles.

Cette couche d'activation est par la suite concaténée à la sortie de la représentation vectorielle des entrées du modèle de classification binaire.

3.3 Le modèle récurrent à l'échelle des mots

En complément du traitement fait pour les caractères, nous avons ajouté un paradigme similaire opérant au niveau des mots afin de pouvoir prendre en compte des dépendances plus lointaines et des contraintes plus fortes, notamment syntaxiques, dans notre système de détection d'erreurs. Ce modèle se voit donc doté d'une structure similaire à celle établie pour les symboles. La principale différence résidant dans le fait que celui-ci doit inclure les informations déduites au niveau des caractères dans son traitement de détection d'erreurs sur les mots. Pour cela, nous ajoutons à cette partie du système une entrée sous la forme d'une représentation issue de la couche d'activation de la détection de symboles erronés. Ce système que nous avons utilisé lors de la compétition peut être schématisé par la Figure 1.

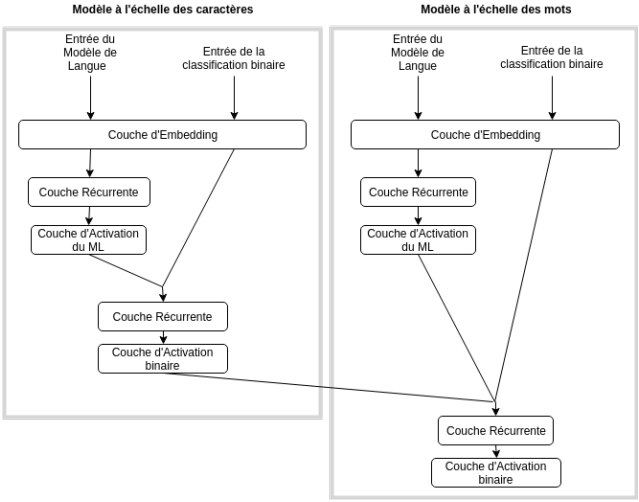


FIGURE 1: Schématisation du système combinant la représentation à l'échelle des caractères et des mots.

4 Configuration expérimentale

Pour entraîner le système ayant participé à la compétition ICDAR Nous avons utilisé $\frac{2}{3}$ de la totalité du corpus que nous avions à disposition (tableau 1) comme corpus d'entraînement et les $\frac{1}{3}$ restant comme corpus de validation. Les modèles de langue à l'échelle des caractères mais aussi à celle des mots ont étaient entraînés sur la portion du corpus de la compétition réservée à cet effet, ainsi que sur des textes issus de dépêches AFP des années passées afin d'accroître le vocabulaire et de permettre au ML de pouvoir effectuer de meilleures généralisations. Nous avons ajouté ces données à hauteur de 25 000 nouveaux termes pour chaque langage (français et anglais), ceci semblant être le seuil optimal pour lequel l'ajout de ces éléments de lexique puisse apporter des gains, avant d'engendrer une dégradation des performances sur le corpus d'évaluation. Notre modèle utilise dans sa version finale un lexique de 256 symboles caractères et de 136 752 mots, incluant la ponctuation.

Il est important de noter que la détection d'erreurs est un problème très déséquilibré au niveau de la représentation des données. En effet, ces dernières représentent, dans le corpus mis à notre disposition durant la compétition, approximativement 2% de la totalité des mots. Un réseau de neurones entraîné sur une telle collection aura tôt fait de toujours choisir comme résultat de prédiction la catégorie d'exemples dominante, c'est-à-dire la classe *non-erronée*. Pour surmonter cette difficulté, nous avons, durant la post-propagation ayant lieu lors de l'apprentissage, doublé le poids attribué par la fonction de coût lorsqu'une mauvaise prédiction devant être attribuée à la classe d'erreur se présentait. Cela dans le but d'augmenter l'impact des corrections faites au modèle durant cette phase.

5 Résultats et évaluation

La tache de détection d'erreurs de la compétition Post OCR Text Correction est évaluée à l'échelle des tokens, qui ne sont autres, pour les organisateurs, qu'une suite de symboles séparés par un espacement

(incluant les tabulations et les caractères vides), ponctuation comprise. Les métriques utilisées sont le *Rappel*, la *Précision* et la *F-mesure*, avec un classement des participants effectué sur la *F-mesure*.

Le jeu de données de test contient 177K mots (96,5K en anglais et 80,6K en français) avec des taux d'OCR bien plus élevés que pour les corpus d'entraînement (de 7 à 10%).

Comme on peut le voir dans le tableau 2, les résultats officiels de la compétition ICDAR (Chiron *et al.*, 2017a) placent notre système à la seconde position dans trois des quatre catégories de textes. Les résultats obtenus sont stables pour la langue anglaise indépendamment du type de document et suivent la même chute de performances lorsque nous comparons les scores de nos sorties à celles des gagnants de cette compétition sur la partie du corpus attribuée aux textes français, en particuliers les monographies comme décrit dans le Tableau 2. Ceci peut être en partie expliqué par le fait que le taux d'erreur est moindre sur le corpus de textes français, les erreurs étant dès lors plus difficiles à mettre en évidence.

	Mono-EN	**Perio-EN**	**Mono-FR**	**Perio-FR**	**Confondus**
A	0.73	0.68	0.55	0.69	0.66
X	**0.66**	**0.66**	**0.43**	**0.60**	0.58
B	0.69	0.54	0.40	0.54	0.54
C	0.67	0.64	0.31	0.50	0.53
D	0.66	0.44	0.36	0.41	0.46

TABLE 2: Résultats officiels pour la tâche de détection en F-mesure. Notre système y est noté **X**.

Nous avons généré des résultats comparatifs durant la période de la compétition afin de valider notre approche en deux étapes sur le corpus de développement. Ces résultats sont regroupés dans le Tableau 3. **C-RNN** fait référence au système de classification binaire à l'échelle des symboles, **C-ML** le Modèle de Langue à l'échelle des caractères, **M-RNN** et **M-ML** désignent les mêmes types de systèmes, à l'échelle des mots. Comme nous pouvons le voir, le modèle **C-RNN** obtient la meilleure précision, mais un rappel très faible lorsqu'il est entraîné seulement sur les données du corpus de la compétition. L'ajout du ML accompagné de données additionnelles provenant des dépêches AFP augmente fortement le rappel. Les modèles à l'échelle des mots, sont quant à eux plus robustes. Cependant les combiner au travers d'un unique modèle permet un gain global des performances.

Modèles	**F-mesure**	**Rappel**	**Précision**
C-RNN	0.24	0.14	**0.62**
C-RNN + C-ML	0.45	0.42	0.48
M-RNN + M-ML	0.53	0.65	0.45
Modèle final	**0.55**	**0.67**	0.46

TABLE 3: Résultats comparatifs dépendemment du modèle utilisé

6 Conclusion

Nous avons présenté dans ce papier un système de détection d'erreurs pour des textes issus de sorties d'OCR dans les langues anglaise et française, développé pour la compétition Post-OCR Text Correction ayant eu lieu lors de la conférence ICDAR-2017. Afin de détecter les possibles erreurs de

telles sorties, nous avons proposé une approche basée sur l'analyse à différents niveau d'informations textuelles dans un réseau de neurones récurrent. La première partie de ce système est entraînée à l'échelle des caractères et la seconde à celle des mots accompagnés d'informations alignées sur chaque mot provenant du modèle à l'échelle des symboles. Cette architecture de classification pour la détection d'erreurs a obtenu de bons résultats lors de l'évaluation de la compétition ICDAR 2017 et notre système a été classé second parmi 11 participants.

Références

ABDULKADER A. & CASEY M. R. (2009). Low cost correction of ocr errors using learning in a multi-engine environment. In *Document Analysis and Recognition, 2009. ICDAR'09. 10th International Conference on*, p. 576–580 : IEEE.

BASSIL Y. & ALWANI M. (2012). Ocr post-processing error correction algorithm using google online spelling suggestion. *arXiv preprint arXiv :1204.0191*.

BÉCHET F. & FAVRE B. (2013). Asr error segment localization for spoken recovery strategy. In *Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on*, p. 6837–6841 : IEEE.

BÉRARD A., PIETQUIN O. & BESACIER L. (2017). Lig-cristal system for the wmt17 automatic post-editing task. *arXiv preprint arXiv :1707.05118*.

BOKSER M. (1992). Omnidocument technologies. *Proceedings of the IEEE*, **80**(7), 1066–1078.

CHIRON G., DOUCET A., COUSTATY M., VISANI M. & MOREUX J.-P. (2017a). ICDAR2017 competition on post-OCR text correction. In *Proceedings of the 14th IAPR International Conference on Document Analysis and Recognition (ICDAR2017)* : IEEE.

CHIRON G., DOUCET A., COUSTATY M., VISANI M. & MOREUX J.-P. (2017b). Impact of ocr errors on the use of digital libraries : Towards a better access to information. In *Digital Libraries (JCDL), 2017 ACM/IEEE Joint Conference on*, p. 1–4 : IEEE.

CHO K., VAN MERRIËNBOER B., GULCEHRE C., BAHDANAU D., BOUGARES F., SCHWENK H. & BENGIO Y. (2014). Learning phrase representations using rnn encoder-decoder for statistical machine translation. *arXiv preprint arXiv :1406.1078*.

CHOLLET F. *et al.* (2015). Keras. https ://github.com/fchollet/keras.

EVERSHED J. & FITCH K. (2014). Correcting noisy ocr : Context beats confusion. In *Proceedings of the First International Conference on Digital Access to Textual Cultural Heritage*, p. 45–51 : ACM.

KOLAK O. & RESNIK P. (2002). Ocr error correction using a noisy channel model. In *Proceedings of the second international conference on Human Language Technology Research*, p. 257–262 : Morgan Kaufmann Publishers Inc.

LIBOVICKÝ J., HELCL J., TLUSTÝ M., PECINA P. & BOJAR O. (2016). Cuni system for wmt16 automatic post-editing and multimodal translation tasks. *arXiv preprint arXiv :1606.07481*.

SCHULZ S. & KUHN J. (2017). Multi-modular domain-tailored ocr post-correction. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, p. 2706–2716.

SMITH R. (2011). Limits on the application of frequency-based language models to ocr. In *Document Analysis and Recognition (ICDAR), 2011 International Conference on*, p. 538–542 : IEEE.

Le benchmarking de la reconnaissance d'entités nommées pour le français

Jungyeul Park
CONJECTO. 74 rue de Paris, 35000 Rennes, France
http://www.conjecto.com

RÉSUMÉ
Cet article présente une tâche du benchmarking de la reconnaissance de l'entité nommée (REN) pour le français. Nous entrainons et évaluons plusieurs algorithmes d'étiquetage de séquence, et nous améliorons les résultats de REN avec une approche fondée sur l'utilisation de l'apprentissage semi-supervisé et du reclassement. Nous obtenons jusqu'à 77.95%, améliorant ainsi le résultat de plus de 34 points par rapport du résultat de base du modèle.

ABSTRACT
Benchmarking for French NER.

This paper presents a benchmarking task of named-entity recognition for French. We train and evaluate several sequence labeling algorithms, and we improve named-entity recognition results using semi-supervised learning and reranking. We obtain up to 77.95%, in which we improve the result by over 34 points compared to the baseline results.

MOTS-CLÉS : Reconnaissance d'entités nommées, REN, benchmarking, évaluation, français.

KEYWORDS: Named-entity recognition, NER, benchmarking, evaluation, French.

1 Named Entity Recognition

Named entities are phrases that contain the names of persons, organizations and locations (Tjong Kim Sang & De Meulder, 2003). The task of named-entity recognition (NER) seeks to identify elements into predefined categories such as the names of persons (PER), locations (LOC), organizations (ORG), etc. The following example [1] is from CoNLL 2003 English NER data :

U.N.	NNP	I-ORG
official	NN	O
Ekeus	NNP	I-PER
heads	VBZ	O
for	IN	O
Baghdad	NNP	I-LOC
.	.	O

In this example, entities such as PER, LOC and ORG are tagged using the BIO format alongside their

1. The example excerpted from https://www.clips.uantwerpen.be/conll2003/ner

original :	preprocessed :
Emmanuel I-PER	Emmanuel NAM I-PER
DESOLES I-PER	DESOLES NAM I-PER
de O	de PRP O
LOU O	LOU NAM O
Directeur O	Directeur NAM O
politique O	politique ADJ O
BÊ>ÀCTION O	BÊ>ÀCTION NAM O
ET O	ET NAM O
ADMINISTRATION O	ADMINISTRATION NAM O
9& O	9& ADJ O
, O	, PUN O
Rue I-LOC	Rue NOM I-LOC
du I-LOC	du PRP:det I-LOC
Pré-Botté I-LOC	Pré-Botté NAM I-LOC
, O	, PUN O
aS O	aS VER:simp O
RENNES I-LOC	RENNES NAM I-LOC
ABONNEMENTS O	ABONNEMENTS NAM O
Dép O	Dép NAM O
. O	. SENT O

FIGURE 1 – Original and preprocessed NER data for French

words and Penn tagset part-of-speech (POS) labels. B-I-O stands for beginning-inside-outside of each entity.

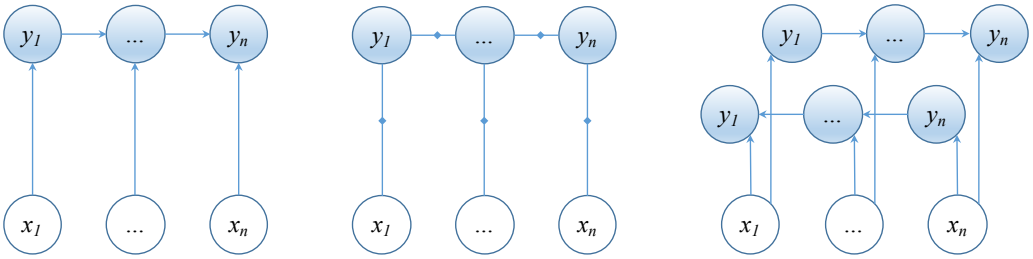
This paper presents a benchmarking task for French NER. We train and evaluate several sequence labeling algorithms such as a Hidden Markov model (HMM) (Rabiner, 1989), conditional random fields (CRF) (Lafferty *et al.*, 2001), and bi-directional long-short-term-memory recurrent neural network (bi-LSTM RNN) (Graves & Schmidhuber, 2005) for French NER. We also improve NER results by introducing semi-supervised learning in which we use a large monolingual corpus to augment the training data, and reranking which adjusts the results based on several sequence labeling algorithms.

2 Experiments and Results

2.1 Data

We use the French NER data provided by Europeana Newspapers [2]. They are OCRed newspaper from 1870 to 1939 taken from the National Library of France. The original data only provides automatically tokenized text and named entity label for each token. There are no sentence boundaries. For training and evaluation, we add "rough" sentence boundaries and POS labels by `TreeTagger` (Schmid, 1994) [3]. To the best of author's knowledge, there are no previous results on this corpus. We explicitly introduce sentence boundaries that machine learning algorithms are trained sentence by sentence based on the `TreeTagger` sentence segmentation. We then split the corpus 80/10/10 ratio as training/development/test data sets, and it gives 10,041/1,255/1,255 sentences, respectively. Figure 1 shows the original data and preprocessed NER data for French. Note that the present corpus is "original". If there may be errors, it is not corrected in this paper.

2. Available at `https://github.com/EuropeanaNewspapers/ner-corpora`
3. Available at `http://www.cis.uni-muenchen.de/~schmid/tools/TreeTagger/`



(a) HMM (b) CRFs (c) bi-LSTM RNN

FIGURE 2 – Learning models for NER : figures for HMM and CRFs are inspired by Sutton & McCallum (2012).

2.2 Learning models

We use following learning models to train and evaluate NER for French :

— HMM using `TnT` (Brants, 2000) [4]
— CRFs using `CRF++` [5]
— CRFs using `Wapiti` (Lavergne *et al.*, 2010) [6]
— bi-LSTM RNN using `NeuroNER` (Dernoncourt *et al.*, 2017) [7] with a pre-trained embedding vector for French (Bojanowski *et al.*, 2017) [8].

Figure 2 summarizes the learning models of HMM, CRFs and bi-LSTM RNN where x_i is a word and y_i is a label ($1 \leq i \leq n$). While an HMM uses only the token's observation probability and the transition probability of states (label) for learning features, CRFs can use their features as we define. We use ± 2 word and POS window context information and a bi-gram word and POS model are used as a feature set for CRFs. The neural network will learn the optimal features during training for the bi-LSTN RNN. We run the experiment with 50 epochs with stochastic gradient descent (SGD), 0.005 learning rate, and 0.5 dropout rate . A pre-trained embedding vector for French (Bojanowski *et al.*, 2017) is in 300 dimensional space, and it enriches word vector results with subword information.

2.3 Results

We evaluate NER results with the standard F_1 metric using `conlleval` [9]. Table 1 shows the overall baseline results on NER for French using several sequence labeling algorithms. Note that we use train+dev data for training for `TnT` and `CRF++` because they cannot have development data during training. Training without dev data can obtain 45.36% and 63.16% for `TnT` and `CRF++`, respectively, which are being outperformed by training with train+dev data as we present in Table 1. Otherwise, we use train/dev/evaluation data as described in §2.1. crf (w) can improve up to 65.95% if the L2 penalty parameter for ridge regression is set λ to 0.01, which can penalize the high-value weights to

4. Available at `http://www.coli.uni-saarland.de/~thorsten/tnt/`
5. Available at `https://taku910.github.io/crfpp/`
6. Available at `https://wapiti.limsi.fr/`
7. Available at `http://neuroner.com/`
8. Available at `https://fasttext.cc/docs/en/pretrained-vectors.html`
9. Available at `https://www.clips.uantwerpen.be/conll2003/ner/`

	hmm (t)	crf (+)	crf (w)	bi-lstm
precision	38.99	58.49	60.13	73.71
recall	55.37	72.06	73.01	78.99
F_1	45.76	64.57	65.38	76.26

TABLE 1 – Overall baseline results on NER for French : crf (+) and crf (w) represent CRFs using `CRF++` and `Wapiti`, respectively.

avoid overfitting. We also note that results on CRFs can be improved using the different feature set. Even though `CRF++` and `Wapiti` implement the same algorithm, `Wapiti` gives the better results. We assume that this is because stop criteria of implementations and default values that we use for learning. [10] While bi-LSTM RNN improves up to 77.76% during training epochs, we present the best result based on dev data.

3 Improving NER Models Using Semi-supervised Learning

We employ the NER model described in the previous section (§2.3) to improve NER results using semi-supervised learning, in which we automatically annotate a large monolingual corpus. This kind of practice is often called self-training (McClosky *et al.*, 2006a), self-taught learning (Raina *et al.*, 2007), and lightly-supervised training (Schwenk, 2008). For semi-supervised learning we introduce the consensus method $\hat{\mathcal{D}}$ (Brodley & Friedl, 1999). We use it by intersection between entity-annotated results using

$$\hat{\mathcal{D}} = \mathcal{D}(\mathcal{M}_1) \cap \cdots \cap \mathcal{D}(\mathcal{M}_n) \tag{1}$$

where $\mathcal{D}$ is raw text data, $\mathcal{M}_i$ is a learning model to annotate raw text data ($1 \leq i \leq n$), and $\hat{\mathcal{D}}$ is filtered annotated data. For raw text data for French, we use the monolingual corpus from the French treebank (Abeillé *et al.*, 2003) [11] (sentences only), and the French News Commentary v10 corpus [12]. We directly use morphologically segmented tokens in the treebank, and the preprocessing tools of Moses (Koehn *et al.*, 2007) for the new commentary corpus : normalizing punctuations and tokenization. [13] Table 2 summarizes the size of the monolingual corpus. To present the characteristics of the monolingual corpus, we provide the ratio of entity labels comparing to `per` in $\hat{\mathcal{D}}$, in which `per` is the most frequent entity in the original corpus. For example, the original NER training data set (train) contains 4,977 `per` and 4,432 `loc` entities, in which we represent 0.89 for `loc`. Note that the number and the ratio of entities in the French treebank and the New Commentary corpora are based on the automatically labeled entities ($\hat{\mathcal{D}}$).

Table 3 shows the overall results on NER using semi-supervised learning. Since hmm (t) gives the weakest results in the previous section, we exclude it for data intersection. Therefore, we obtain $\hat{\mathcal{D}}$ only from $\mathcal{D}(\mathcal{M}_{crf(+)}) \cap \mathcal{D}(\mathcal{M}_{crf(w)}) \cap \mathcal{D}(\mathcal{M}_{bilstm})$ for the current semi-supervised learning task. All learning algorithms can improve the NER results using semi-supervised learning by benefiting from the larger training data, even though they are automatically labeled. Such improvements using "self-

10. We would like to thank reviewer #3 for indicating this problem.
11. Available at `http://www.llf.cnrs.fr/Gens/Abeille/French-Treebank-fr.php`
12. Available at `http://www.statmt.org/wmt15/training-parallel-nc-v10.tgz`
13. Available at `http://www.statmt.org/moses`

	size ($\mathcal{D}$)	size ($\hat{\mathcal{D}}$)	per	loc	org
(original train)	0.16 M	-	1	0.89	0.42
French treebank	0.62 M	0.38 M	1	0.74	0.18
News Commentary	6.09 M	3.65 M	1	3.00	0.23

TABLE 2 – Size of the monolingual corpus and the ratio of entity labels

	hmm (t)	crf (+)	crf (w)	bi-lstm
French treebank	50.34	65.94	66.63	77.49
News Commentary	49.69	66.18	68.28	76.65

TABLE 3 – Overall results (F_1) on NER for French using semi-supervised learning described in §3

training" have already been shown in many NLP tasks, for example in syntactic parsing (McClosky *et al.*, 2006a).

4 Improving Results Using Reranking

We also propose a reranking algorithm using $\hat{\mathcal{L}} = \mathrm{rerank}(\mathcal{L}_1, ..., \mathcal{L}_n)$ where $\mathcal{L}_i$ is an assigned label by a learning algorithm, and $\hat{\mathcal{L}}$ is a reranked label by the rerank function. We exclude $\mathcal{L}_{hmm(t)}$, and we then obtain $\hat{\mathcal{L}}$ from $\mathrm{rerank}(\mathcal{L}_{crf(+)}, \mathcal{L}_{crf(w)}, \mathcal{L}_{bilstm})$ for reranking labels. We calculate the rerank function as follows :

$$rerank(\cdot) \quad = \quad \begin{array}{ll} \mathrm{argmax}(\texttt{per}, \texttt{loc}, \texttt{org}) & \text{if there is } \textit{any} \text{ entity label} \\ \texttt{O} & \text{otherwise} \end{array}$$

For each entity score (per, loc or org), we calculate

$$\alpha_1 \mathcal{L}_{crf(+)} + \alpha_2 \mathcal{L}_{crf(w)} + \alpha_3 \mathcal{L}_{bilstm} \tag{2}$$

where α_i is a normalized weight. For example, α_1 for $\mathcal{L}_{crf(+)}$ is calculated by its baseline result in F_1 being normalized by the sum of all F_1 scores by learning algorithms : $\frac{64.57}{64.57+65.38+76.26}$. We use $\alpha_1 = 0.3131$, $\alpha_2 = 0.3171$ and $\alpha_3 = 0.3698$. For example, if a word *Loiret* (a department name in north-central France) is annotated as I-LOC, I-ORG and I-LOC by CRFs and bi-lstm, $\hat{\mathcal{L}}$ is I-LOC by the rerank calculation described in Figure 3. Finally, Table 4 shows the reranking results on NER for French.

$$\begin{array}{lll} \texttt{per} = & \alpha_1 \times 0 + \alpha_2 \times 0 + \alpha_3 \times 0 & = 0 \\ \texttt{loc} = & \alpha_1 \times 1 + \alpha_2 \times 0 + \alpha_3 \times 1 & = 0.6829 \\ \texttt{org} = & \alpha_1 \times 0 + \alpha_2 \times 1 + \alpha_3 \times 0 & = 0.3171 \end{array}$$

FIGURE 3 – An example for the rerank function to calculate $\mathrm{argmax}(\texttt{per}, \texttt{loc}, \texttt{org})$ for *Loiret*. We use $\alpha_1 = 0.3131$, $\alpha_2 = 0.3171$ and $\alpha_3 = 0.3698$.

		monolingual corpus
reranking based on Table 1	76.41	baseline
reranking + semi-supervised based on Table 3	77.95	French treebank
reranking + semi-supervised based on Table 3	77.03	News Commentary

TABLE 4 – Reranking results (F_1) on NER for French described in §4

5 Previous Work

Ollagnier *et al.* (2014) used the Open Edition corpus the Quaero Broadcast News Extended Named Entity corpus [14], which contains over 1.2M tokens. They evaluated NER results with LIA_NE (HMM-CRFs) [15], OpenNLP (Maximum entropy) [16] and Standford NER (CRFs) [17] with different sizes of training data. They obtained up to 57,9 F_1 score with LIA_NE. Partalas *et al.* (2016) compared NER systems in the e-Commerce domain for the cosmetics products by using handcrafted rules and machine learning techniques. They used two 50K tokens data sets (cosmetics magazines and blog articles). They presented only entity level results and a system of lexical combined syntactic rules with a domain-specific dictionary usually outperformed CRFs. Their rule-based systems yielded between 60.00 and 90.68 F_1 scores based on different entities.

There were efforts to create corpora annotated in named entities for French. Sagot *et al.* (2012) and Dutrey *et al.* (2012) manually annotated named entities in the French treebank, and in restricted domain such as oral dialogs recored by the EDF call center for information extraction, respectively. Okinina *et al.* (2013) enriched proper nouns by mining Wikipedia with the combination of DBpedia rules and a support vector machine classification. Hatmi (2012) used a cross-lingual approach by converting a rule-based English NER system into French by using lexical and grammar adaptations.

Fraisse *et al.* (2013) employed NER for better classification results on opinion mining and sentiment analysis. Sagot & Gábor (2014) corrected OCRed named entities errors by using a rule-based NER system. Brando *et al.* (2016) used NER for recognizing geographical references. These are applications, in which NER results improved other natural language processing tasks. Otherwise, Dupont & Tellier (2014) proposed a pipeline for French NER based on `Wapiti`.

6 Conclusion

In this paper, we trained and evaluated several sequence labeling algorithms to perform benchmarking for French named-entity recognition data. We then improved NER results using semi-supervised learning and reranking. We obtained up to 77.95%, in which we improved the result by over 34 points compared to the baseline results of the HMM.

While incorporating unlabeled data into a new model is a simple method, it would not be surprising that self-training is not normally effective because errors in the original model can be amplified in

14. Available at `http://catalog.elra.info/product_info.php?products_id=1195`
15. Available at `http://pageperso.lif.univ-mrs.fr/~frederic.bechet`
16. Available at `https://opennlp.apache.org`
17. Available at `https://nlp.stanford.edu/software/CRF-NER.shtml`

the new model (McClosky *et al.*, 2006a). We discard the weakest learner's results for the consensus method. This decision actually improves the NER results. For example, while hmm (t) obtains only 47.35% with intersection of all data for the French treebank, it achieve 50.34% by excluding $\mathcal{D}(\mathcal{M}_{hmm(t)})$ for data intersection. This semi-supervised process can be iterated, and it can be performed over other sets of unlabeled data for French. We assume that iterating the semi-supervised process and using a larger unlabeled data can improve NER results. We leave this to future work. However, while learning models for HMM and CRFs are relatively quick, we note that training bi-lstm using a large annotated corpus (*e.g.* over 3.65M tokens in News Commentary) takes several days even on a GPU for a single iteration.

Reranking basically selects the best result from the set of NER results for each sentence to have constructed high-performance NLP systems such as parsing (Charniak & Johnson, 2005). Combining reranking and self-training is not new, which has been, for example, already proposed for syntactic parsing (McClosky *et al.*, 2006b). While reported results show a minor improvement (*e.g.* we obtain 76.41% using ranking baseline, compared to 76.26% in the best baseline result), it is cheap and easy to implement for immediate improvements.

Comparison of results using previously proposed NER systems for French, and benchmark learning using other previously proposed NER data would be an interesting task, and we leave this to future work. All trained models and data will be publicly available at `https://github.com/jungyeul/taln2018`.

Remerciements

We would like to thank the anonymous reviewers for their suggestions and comments.

Références

ABEILLÉ A., CLÉMENT L. & TOUSSENEL F. (2003). Building a Treebank for French. In *Treebanks*, p. 165–188. Kluwer.

BOJANOWSKI P., GRAVE E., JOULIN A. & MIKOLOV T. (2017). Enriching Word Vectors with Subword Information. *Transactions of the Association for Computational Linguistics*, **5**, 135–146.

BRANDO C., DOMINGUÈS C. & CAPEYRON M. (2016). Evaluation of NER Systems for the Recognition of Place Mentions in French Thematic Corpora. In *Proceedings of the 10th Workshop on Geographic Information Retrieval*, GIR '16, p. 7 :1–7 :10, New York, NY, USA : ACM.

BRANTS T. (2000). TnT – A Statistical Part-of-Speech Tagger. In *Proceedings of the Sixth Conference on Applied Natural Language Processing*, p. 224–231, Seattle, Washington, USA : Association for Computational Linguistics.

BRODLEY C. E. & FRIEDL M. A. (1999). Identifying Mislabeled Training Data. *Journal of Artificial Intelligence Research*, **11**, 131–167.

CHARNIAK E. & JOHNSON M. (2005). Coarse-to-Fine n-Best Parsing and MaxEnt Discriminative Reranking. In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL'05)*, p. 173–180, Ann Arbor, Michigan : Association for Computational Linguistics.

DERNONCOURT F., LEE J. Y. & SZOLOVITS P. (2017). NeuroNER : an easy-to-use program for named-entity recognition based on neural networks. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing : System Demonstrations*, p. 97–102, Copenhagen, Denmark : Association for Computational Linguistics.

DUPONT Y. & TELLIER I. (2014). Un reconnaisseur d'entités nommées du Français. In *Proceedings of TALN 2014 (Volume 3 : System Demonstrations)*, p. 40–41, Marseille, France : Association pour le Traitement Automatique des Langues.

DUTREY C., CLAVEL C., ROSSET S., VASILESCU I. & ADDA-DECKER M. (2012). Quel est l'apport de la détection d'entités nommées pour l'extraction d'information en domaine restreint ? (What is the contribution of named entities detection for information extraction in restric- ted domain ?) [in French]. In *Proceedings of the Joint Conference JEP-TALN-RECITAL 2012, volume 2 : TALN*, p. 359–366, Grenoble, France : ATALA/AFCP.

FRAISSE A., PAROUBEK P. & FRANCOPOULO G. (2013). L'apport des Entités Nommées pour la classification des opinions minoritaires. In *Proceedings of TALN 2013 (Volume 2 : Short Papers)*, p. 588–595, Les Sables d'Olonne, France : ATALA.

GRAVES A. & SCHMIDHUBER J. (2005). Framewise phoneme classification with bidirectional LSTM and other neural network architectures. *Neural Networks*, **18**(5-6), 602–610.

HATMI M. (2012). Adaptation d'un système de reconnaissance d'entités nommées pour le français à l'anglais à moindre coût (Adapting a French Named Entity Recognition System to English with Minimal Costs) [in French]. In *Proceedings of the Joint Conference JEP-TALN-RECITAL 2012, volume 3 : RECITAL*, p. 151–161, Grenoble, France : ATALA/AFCP.

KOEHN P., HOANG H., BIRCH A., CALLISON-BURCH C., FEDERICO M., BERTOLDI N., COWAN B., SHEN W., MORAN C., ZENS R., DYER C., BOJAR O., CONSTANTIN A. & HERBST E. (2007). Moses : Open Source Toolkit for Statistical Machine Translation. In *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics Companion Volume Proceedings of the Demo and Poster Sessions*, p. 177–180, Prague, Czech Republic : Association for Computational Linguistics.

LAFFERTY J. D., MCCALLUM A. & PEREIRA F. C. N. (2001). Conditional Random Fields : Probabilistic Models for Segmenting and Labeling Sequence Data. In *Proceedings of the Eighteenth International Conference on Machine Learning*, ICML '01, p. 282–289, San Francisco, CA, USA : Morgan Kaufmann Publishers Inc.

LAVERGNE T., CAPPÉ O. & YVON F. (2010). Practical Very Large Scale CRFs. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, p. 504–513, Uppsala, Sweden : Association for Computational Linguistics.

MCCLOSKY D., CHARNIAK E. & JOHNSON M. (2006a). Effective Self-Training for Parsing. In *Proceedings of the Human Language Technology Conference of the NAACL, Main Conference*, p. 152–159, New York City, USA : Association for Computational Linguistics.

MCCLOSKY D., CHARNIAK E. & JOHNSON M. (2006b). Reranking and Self-Training for Parser Adaptation. In *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics*, p. 337–344, Sydney, Australia : Association for Computational Linguistics.

OKININA N., NOUVEL D., FRIBURGER N. & ANTOINE J.-Y. (2013). Supervised learning on encyclopaedic resources for the extension of a lexicon of proper names dedicated to the recognition of named entities (Apprentissage supervisé sur ressources encyclopédiques pour l'enrichissement

d'un lexique de noms propres destiné. In *Proceedings of TALN 2013 (Volume 2 : Short Papers)*, p. 667–674, Les Sables d'Olonne, France : ATALA.

OLLAGNIER A., FOURNIER S., BELLOT P. & BÉCHET F. (2014). Impact of the nature and size of the training set on performance in the automatic detection of named entities (Impact de la nature et de la taille des corpus d'apprentissage sur les performances dans la détection automatique des entités nommées) [in Frenc. In *Proceedings of TALN 2014 (Volume 2 : Short Papers)*, p. 511–516, Marseille, France : Association pour le Traitement Automatique des Langues.

PARTALAS I., LOPEZ C. & SEGOND F. (2016). Comparing Named-Entity Recognizers in a Targeted Domain : Handcrafted Rules vs. Machine Learning. In *Actes de la conférence conjointe JEP-TALN-RECITAL 2016, volume 2 : TALN*, p. 389–395, Paris, France : Association pour le Traitement Automatique des Langues.

RABINER L. R. (1989). A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition. *Proceedings of the IEEE*, **77**(2), 257–286.

RAINA R., BATTLE A., LEE H., PACKER B. & NG A. Y. (2007). Self-taught Learning : Transfer Learning from Unlabeled Data. In *Proceedings of the 24th International Conference on Machine Learning*, ICML '07, p. 759–766, New York, NY, USA : ACM.

SAGOT B. & GÁBOR K. (2014). Détection et correction automatique d'entités nommées dans des corpus OCRisés. In *Proceedings of TALN 2014 (Volume 2 : Short Papers)*, p. 437–442, Marseille, France : Association pour le Traitement Automatique des Langues.

SAGOT B., RICHARD M. & STERN R. (2012). Annotation référentielle du Corpus Arboré de Paris 7 en entités nommées (Referential named entity annotation of the Paris 7 French TreeBank) [in French]. In *Proceedings of the Joint Conference JEP-TALN-RECITAL 2012, volume 2 : TALN*, p. 535–542, Grenoble, France : ATALA/AFCP.

SCHMID H. (1994). Probabilistic Part-of-Speech Tagging Using Decision Trees. In *Proceedings of International Conference on New Methods in Language Processing*, Manchester, UK.

SCHWENK H. (2008). Investigations on Large-Scale Lightly-Supervised Training for Statistical Machine Translation. In *Proceedings of the International Workshop on Spoken Language Translation*, p. 182–189, Hawaii, USA.

SUTTON C. & MCCALLUM A. (2012). An Introduction to Conditional Random Fields. *Foundations and Trends® in Machine Learning*, **4**(4), 267–373.

TJONG KIM SANG E. F. & DE MEULDER F. (2003). Introduction to the CoNLL-2003 Shared Task : Language-Independent Named Entity Recognition. In W. DAELEMANS & M. OSBORNE, Eds., *Proceedings of the Seventh Conference on Natural Language Learning at HLT-NAACL 2003*, p. 142–147.

Une note sur l'analyse du constituant pour le français

Jungyeul Park
CONJECTO, 74 rue de Paris, 35000 Rennes, France
http://www.conjecto.com

RÉSUMÉ

Cet article traite des analyses d'erreurs quantitatives et qualitatives sur les résultats de l'analyse syntaxique des constituants pour le français. Pour cela, nous étendons l'approche de Kummerfeld *et al.* (2012) pour français, et nous présentons les détails de l'analyse. Nous entraînons les systèmes d'analyse syntaxique statistiques et neuraux avec le corpus arboré pour français, et nous évaluons les résultats d'analyse. Le corpus arboré pour le français fournit des étiquettes syntagmatiques à grain fin, et les caractéristiques grammaticales du corpus affectent des erreurs d'analyse syntaxique.

ABSTRACT

A Note on constituent parsing for French.

This paper deals with the quantitative and qualitative error analysis on French constituent parsing results. To this end, we extend the approach of Kummerfeld *et al.* (2012) to the French treebank for parser error analysis, and present details of the analysis for French. We train statistical and neural parsing systems, and evaluate parsing results using the French treebank. The French treebank provides fine-grained phrase labels and grammatical characteristics of the French treebank affect parsing errors.

MOTS-CLÉS : Analyse du constituant, corpus arboré, erreurs d'analyse syntaxique, systèmes d'analyse syntaxique statistiques et neuraux, français.

KEYWORDS: Constituent parsing, treebank, parsing errors, statistical and neural parsing systems, French.

1 Constituent Parsing for French

Treebanks, collections of parsed and syntactically annotated corpora, constitute an essential resource for natural language processing in any given language. The automatic syntactic analysis of sentences directly benefits from syntactically annotated corpora. Currently, most of the state-of-the-art parsers use the statistical or neural parsing approaches. These parsers use annotated syntactic information in the treebank to train parsing models. Several annotated phrase-structured treebanks have been created for French such as the French treebank (Abeillé *et al.*, 2003) and the Sequoia corpus (Candito & Seddah, 2012). Table 1 summarizes previous work on constituent parsing for French. This paper is intended to present several factors on constituent parsing for French including parsing results and an error analysis. We train and evaluate the French treebank (Abeillé *et al.*, 2003) using the state-of-art parsing systems : the statistical Berkeley parser (Petrov *et al.*, 2006) and the neural Trance parser (Watanabe & Sumita, 2015) (§ 2). Then, we extend Kummerfeld *et al.* (2012)'s parser error analysis to French (§ 3). Finally, we conclude the paper with discussion and future perspectives (§ 4).

Seddah *et al.* (2009)	84,93	using the Berkeley parser
Candito & Crabbé (2009)	88.29	gold POS + morphological clustering using Brown clustering
Candito & Seddah (2010)	87.80	gold lemma/POS + morphological clustering
Sigogne *et al.* (2011)	85.22	integrating the Lexicon-Grammar
Le Roux *et al.* (2014)	83.80	recognizing MWEs usnig CRFs and dual decomposition
Durrett & Klein (2015)	81.25	neural CRF parsing for multilingual settings
Coavoux & Crabbé (2016)	80.56	transition-based parsing with dynamic oracle (order-0 head-markovization)
Cross & Huang (2016)	83.31	transition-based parsing with dynamic oracle (no binarization)

TABLE 1 – Brief description and results of previous work on constituent parsing for French : Le Roux *et al.* (2014), Durrett & Klein (2015), Coavoux & Crabbé (2016) and Cross & Huang (2016) are based on a corpus split proposed in Seddah *et al.* (2013).

The main contribution of this paper is as follows. First, we explore various settings to parse the French treebank including parsing with functional information. Secondly, we propose parsing errors analysis for French based on Kummerfeld *et al.* (2012) to present the quantitative and qualitative error analysis. The error analysis script for French is publicly available at `https://github.com/jungyeul/taln2018`.

2 Experiments and Results

The current available version of the French treebank contains 45 files and 21,550 sentences (Abeillé *et al.*, 2003). [1] We use a corpus split proposed in Seddah *et al.* (2013) for training, development and test datasets directly from the French treebank instead of the distribution version from the SPMRL 2013 Shared Task. [2] This is mainly to train/evaluate the treebank using the different annotation such as training with functional information. While there are more sentences in the current treebank with 17,774/1,235/2,541 sentences for training/dev/evaluation, we use the exact data split from (Seddah *et al.*, 2013) (14,759/1,235/2,541). For statistical parsing using the Berkeley parser (Petrov *et al.*, 2006) [3], we report evaluation results using grammars which give the best results on development data. While the original Berkeley parser proposed several runs of training because of the EM algorithm which can find locally maximum likelihood parameters, we empirically found that each run of training gives same results. Therefore, we use the single run of training using the Berkeley parser with the default option. For experiments in this paper, we use Penn treebank-like preprocessing, especially by removing null elements (`*T*`) and functional information in the phrase label (*e.g.* `-SUJ` or `-OBJ`) as described in (Bikel, 2004). We evaluate the parser accuracy with the standard F_1 metric from `EVALB`. [4] While the SPMRL shared task provides the alternative `EVALB` [5], it produces the same F_1 scores for French. We only change the original `evalb` to display results for sentences $\leq$70 as in the

1. `http://www.llf.cnrs.fr/Gens/Abeille/French-Treebank-fr.php`
2. `http://www.spmrl.org/spmrl2013-sharedtask.html`
3. `https://github.com/slavpetrov/berkeleyparser`
4. `http://nlp.cs.nyu.edu/evalb`
5. `http://pauillac.inria.fr/~seddah/evalb_spmrl2013.tar.gz`

	berkeley+r	berkeley+f
(w/o gold POS)	79.26 (81.51)	77.02 (79.59)
(w/ gold POS)	80.95 (83.37)	78.55 (81.25)
# of NT label type	23	111

TABLE 2 – Parsing results using the statistical parser and the number of phrase non-terminal label types. For parsing results we also present F1 scores for sentences $\leq$70 in parentheses.

	trance+r	trance+f
(w/o gold POS)	78.05 (80.77)	76.39 (79.03)

TABLE 3 – Parsing results using the neural parser

shared task. We rename phrase labels which share the same label names with POS labels (usually for multi-word expressions or compound words) (+r). For example, we convert [$_{\mathrm{P}}$ [$_{\mathrm{P}}$ *D'*][$_{\mathrm{P}}$ *après*]] into [$_{\mathrm{P+}}$ [$_{\mathrm{P}}$ *D'*][$_{\mathrm{P}}$ *après*]] to differentiate between `P`s in the phrase label and the POS label. Therefore, we rename POS labels `A`, `ADV`, `C`, `CL`, `D`, `ET`, `I`, `N`, `P`, `PRO`, and `V` which also appear in the phrase labels. We note that the treebank of the SPMRL shared task has a similar annotation for compound words. For comparison reason, we also use functional information during training (+f) without renaming phrase labels. For example `np+suj` and `vppart+mod` instead of `np` and `vppart` are used for (+f). Table 2 shows the current parsing results on evaluation data by the Berkeley parser. Table 2 also shows the number of non-terminal (NT) label type without considering POS labels, in which `berkeley+r` has 12 phrase labels and 11 POS labels (renamed with +). We convert proposed alternative treebank forms (+r and +f) into the original preprocessed form without renaming and functional information to evaluate the result. We present the final scores from evaluation data based on best parsing results of dev data.

For neural parsing, we use the Trance parser (Watanabe & Sumita, 2015) [6] and a pre-trained 300 dimension embedding vector provided by Bojanowski *et al.* (2017) [7]. We use default options with 50 epochs for the Trance parser. Table 2 shows the current parsing results on evaluation data by the Trance parser.

3 Parsing Error Analysis

Recent state of the art parsing techniques are easily trained and evaluated if the syntactically annotated treebank is available. Their results, however, can be difficult to understand because grammars are automatically induced from the treebank. Kummerfeld *et al.* (2012) presented an approach to quantify constituent parsing errors based on the treebank annotation. [8] In this section, we extend Kummerfeld's approach to the French treebank parsing for parser error analysis. Error analysis is based on parsing results (+r). Table 4 shows the quantified number of each error w/o gold POS and w/ gold POS for the Berkeley and the Trance parsers.

6. `https://github.com/tarowatanabe/trance`
7. `https://fasttext.cc/docs/en/pretrained-vectors.html`
8. `https://github.com/jkkummerfeld/berkeley-parser-analyser`

	PP	NP	VP	MD	CL	PR	CO	SW	DL	UN	NI	UD
w/o (B)	2,036	531	380	195	301	17	479	1,885	681	678	444	3,302
w/ (B)	1,953	562	381	171	294	9	673	1,459	435	640	411	3,052
w/o (T)	1,956	593	338	310	282	16	436	1,765	617	728	559	3,841

TABLE 4 – Quantitative error analysis for the Berkeley parser :(B) for the Berkeley parser and (T) for the Trance parser with (w/) and without (w/o) gold POS labels. MD for modifier, CL for clause, PR for pronoun, CO for co-ordination, SW for single word, DL for different label, UN for unary, NI for np internal, and UD for undefined errors.

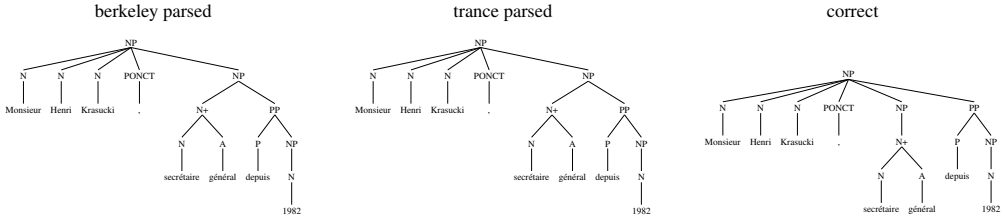


FIGURE 1 – PP attachment error : since pp is wrongly recognized as an argument of the sister np node instead an argument of its parent in (1), pp is low.

Attachment errors Attachment errors are the most frequent errors in constituent parsing for French (over 36% of parsing errors). They generally consist of mistakes and inconsistencies for recognizing arguments of the lexical head. There are six types of attachment errors : pp, np, vp (for vn, vpinf and vppart), modifier (for ap and adp), clause (for sint, srel and ssub), and pron (for cl and pro). See Figure 1 for an example of the PP attachement error.

(1) a. * [NP [N M.] [N Henri] [N Krasucki] [PONCT ,] [NP [N+ [N secrétaire] [A général]] [PP [P depuis] [NP [N 1982]]]]]
b. [NP [N M.] [N Henri] [N Krasucki] [PONCT ,] [NP [N+ [N secrétaire] [A général]]] [PP [P depuis] [NP [N 1982]]]]

Co-ordination error Annotating phrase with co-ordination in French is a difficult problem (*inter alia* Mouret (2007)). The current annotation in the French treebank shows a hierarchical structure, which is different with the English Penn treebank (a flat structure). Finding the correct scope of the coordinating conjunction is challenging, and co-ordination errors occur frequently. See Figure 2 for an example of the co-ordination error.

(2) a. * [PP [P d'] [NP [N ordre] [AP [A économique] [COORD [C et] [AP [A financier]]]]]]
b. * [PP [P d'] [NP [N ordre] [AP [A économique]]]] [COORD [C et] [AP [A financier]]]
c. [PP [P d'] [NP [N [N ordre] [A économique]] [COORD [C et] [AP [A financier]]]]]

Different label A phrase label is wrongly assigned. We note that POS label errors are not counted, and even for parsing with gold POS label, the Berkeley parser does not always obtain 100% for POS labeling accuracy. See Figure 3 for an example of the different label error.

(3) a. * [PP ... [NP [N sommes] [ADV+ [P en] [N jeu]]]]

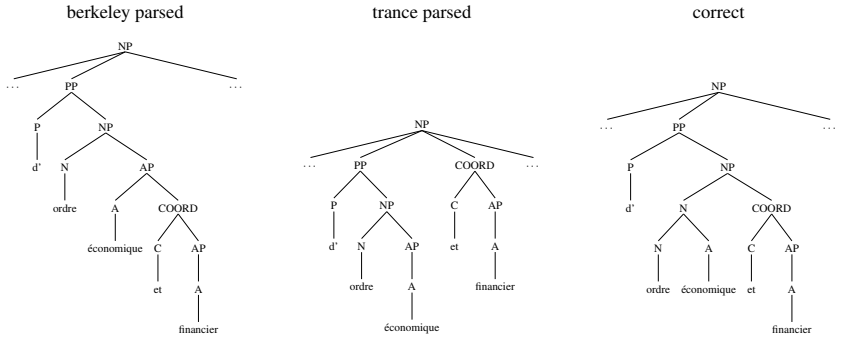


FIGURE 2 – Co-ordination error : `coord` is either low (B) or high (T). A coordinator *et* links with *économique* (B) or *d'ordre économique* (T) in (2).

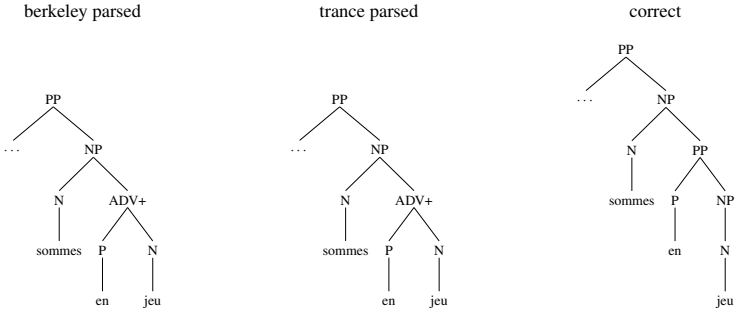


FIGURE 3 – Different label : `adv+` is wrongly recognized for `pp` in (3). It implies another error in which `n` for *jeu* is high (unary error).

b. [PP ... [NP [N sommes] [PP [P en] [NP [N jeu]]]]]

NP internal structure A general structure of the French treebank is relatively flat for the inside of NP as well as the entire sentence. For example, a sentence in (4) is an NP with a flat structure as follows : [NP [D ...] [N+ ...] [AP ...] [PP ...]]. However, both parsers fail to capture the flat structure for NP including a phrase segmentation. See Figure 4 for an example of the NP internal structure error.

(4) a. * [NP [D son] [N droit] [PP [P de] [NP [N préemption] [AP [A possible]]]] [PP [P sur] [NP [D le] [A futur] [N canal] [VPPART [V libéré]]]]

b. * [NP [D son] [N droit] [PP [P de] [NP [N préemption] [AP [A possible]] [PP [P sur] [NP [D le] [A futur] [N+ [N canal] [A libéré]]]]]]

c. [NP [D son] [N+ [N droit] [P de] [N préemption]] [AP [A possible]] [PP [P sur] [NP [D le] [A futur] [N canal] [VPPART [V libéré]]]]

We do not detail single word and unary errors because they are mostly parts of another errors. Over 30% of parsing errors are undefined. We need to investigate these other error types for constituent parsing results, which can be more pertinent for French. We leave this for future work.

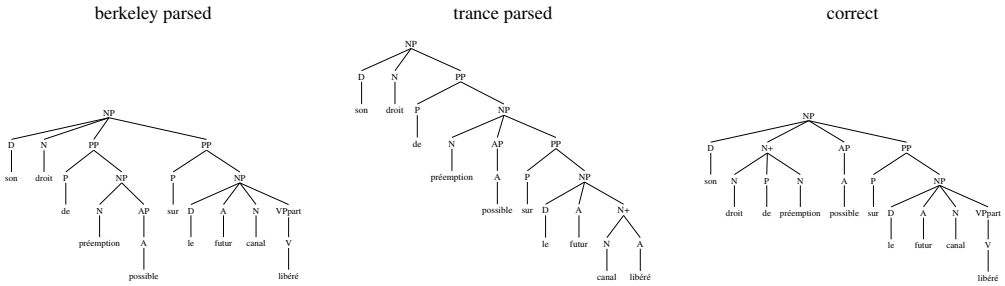


FIGURE 4 – NP internal error : `np` is wrongly constructed (4). It also implies another errors such as MWE recognition for *droit de préemption*, and `ap` (*possible*) and `pp` (*sur le futur ...*) attachment errors.

Previously, Sagot & de la Clergerie (2006) proposed an error mining technique based on parsing results from the FRMG (Thomasset & De la Clergerie, 2005) and the SXLFG (Boullier & Sagot, 2005) parsers. Since they parsed the raw corpus without knowing the correct parsed tree, they tried to find "suspicious" parsed results. These suspicious parsing trees are calculated based on predefined syntactic-related resources such as a morphological and syntactic lexicon Le*fff* (Sagot, 2010) and a pre-syntactic processing pipeline SXPIPE (Sagot & Boullier, 2005).

4 Discussion and Conclusion

This paper dealt with error analysis studies on French constituent parsing results. While a neural parser improved parsing results for other languages such as English and Chinese, we did not obtain the better results for French. There are many intrinsic (learning rate, dropout, # of epochs, etc.) and extrinsic (word embedding and its dimension size) factors. Since training using 50 epochs takes over three or four days to learn a parsing model on a single machine, it wouldn't be easy to find proper parameters for French for neural parsing. We leave finding optimal parameter for French to future work. Functional information would also improve parsing results for certain morphologically rich languages (Chung *et al.*, 2010). The French treebank provides fine-grained phrase labels (111 different labels) and the Berkeley parser also generates additional internal phrase labels during training. PCFG rules in `berkeley+f` contain over 2M, compared to 0.4M in `berkeley+r` (*cf.* 18K vs. 15K for `trance+f` and `trance+r`). Such diversities with phrase labels can give a biased distribution. Therefore, functional information hardly effects or even tends to worsen parsing results in many cases. Investigating the effective way on clustering phrase labels can be one direction to improve parsing results and we leave this for future work. Instead of renaming phrase labels with +, we can also consider renaming with existing $\star$`p`-like labels such as `np` or `pp` : *e.g.* the phrase label of compound words in [$_{\text{NP}}$ [$_{\text{N}}$ [$_{\text{N}}$ *banques*] [$_{\text{A}}$ *centrales*]]] is converted into [$_{\text{NP}}$ [$_{\text{NP}}$ [$_{\text{N}}$ ...] [$_{\text{A}}$ ...]]] instead of [$_{\text{NP}}$ [$_{\text{N+}}$ [$_{\text{N}}$ ...] [$_{\text{A}}$ ...]]]. Using $\star$`p`-like renaming labels (12 phrase labels), we can find additional repetitive unary branches and we remove them during preprocessing. Converting into $\star$`p`-like labels is straightforward except for `D` in which it would be `np` for numbers such as [$_{\text{NP}}$ [$_{\text{D}}$ *vingt*] [$_{\text{PONCT}}$ -] [$_{\text{D}}$ *cinq*] [$_{\text{D}}$ *mille*]] ; otherwise, `pp`. Consequently, we have 284,107 non-terminal nodes instead of 288,374 as in `berkeley+r` (excluding pre-terminal POS labels) in training data, and obtain only up to 75.42% F_1 score. This reflects the fact that recognizing multi-word expressions

(MWEs) and compound words is important in parsing for French, and it already proved in Le Roux *et al.* (2014) where they employed external linguistic resources such as DELAC, compound word dictionary for French (Courtois *et al.*, 1997) [9]. Exploring MWEs can be another direction to improve parsing results. [10] We note that we obtained slightly better results using the Berkeley parser than what the SPMRL shared task reported (gold setting) : 80.38 and 81.76 for w/o and w/ gold POS labels. This is probably because a preprocessing step for treebank data could be "slightly" dissimilar. We used our own the preprocessed French treebank to explore the different treebank settings.

Références

ABEILLÉ A., CLÉMENT L. & TOUSSENEL F. (2003). Building a Treebank for French. In *Treebanks*, p. 165–188. Kluwer.

BIKEL D. M. (2004). Intricacies of Collins' Parsing Model. *Computational Linguistics*, **30**(4), 479–511.

BOJANOWSKI P., GRAVE E., JOULIN A. & MIKOLOV T. (2017). Enriching Word Vectors with Subword Information. *Transactions of the Association for Computational Linguistics*, **5**, 135–146.

BOULLIER P. & SAGOT B. (2005). Efficient and Robust LFG Parsing : SxLFG. In *Proceedings of the Ninth International Workshop on Parsing Technology (IWPT2005)*, p. 1–10, Vancouver, British Columbia : Association for Computational Linguistics.

CANDITO M. & CRABBÉ B. (2009). Improving generative statistical parsing with semi-supervised word clustering. In *Proceedings of the 11th International Conference on Parsing Technologies (IWPT'09)*, p. 138–141, Paris, France : Association for Computational Linguistics.

CANDITO M. & SEDDAH D. (2010). Parsing Word Clusters. In *Proceedings of the NAACL HLT 2010 First Workshop on Statistical Parsing of Morphologically-Rich Languages*, p. 76–84, Los Angeles, CA, USA : Association for Computational Linguistics.

CANDITO M. & SEDDAH D. (2012). Le corpus Sequoia : annotation syntaxique et exploitation pour l'adaptation d'analyseur par pont lexical. In *Proceedings of the Joint Conference JEP-TALN-RECITAL 2012, volume 2 : TALN*, p. 321–334, Grenoble, France : ATALA/AFCP.

CHUNG T., POST M. & GILDEA D. (2010). Factors Affecting the Accuracy of Korean Parsing. In *Proceedings of the NAACL HLT 2010 First Workshop on Statistical Parsing of Morphologically-Rich Languages*, p. 49–57, Los Angeles, CA, USA : Association for Computational Linguistics.

COAVOUX M. & CRABBÉ B. (2016). Neural Greedy Constituent Parsing with Dynamic Oracles. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1 : Long Papers)*, p. 172–182, Berlin, Germany : Association for Computational Linguistics.

COURTOIS B., GARRIGUES M., GROSS G., GROSS M., JUNG R., MATHIEU-COLAS M., MONCEAUX A., ANNE P.-M., SILBERZTEIN M. & VIVÈS R. (1997). *Dictionnaire électronique DELAC : les noms composés binaires*. Rapport interne, Université Paris 7, Paris.

CROSS J. & HUANG L. (2016). Span-Based Constituency Parsing with a Structure-Label System and Provably Optimal Dynamic Oracles. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, p. 1–11, Austin, Texas : Association for Computational Linguistics.

9. http://infolingu.univ-mlv.fr/DonneesLinguistiques/Dictionnaires/delac.html
10. https://typo.uni-konstanz.de/parseme

DURRETT G. & KLEIN D. (2015). Neural CRF Parsing. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1 : Long Papers)*, p. 302–312, Beijing, China : Association for Computational Linguistics.

KUMMERFELD J. K., HALL D., CURRAN J. R. & KLEIN D. (2012). Parser Showdown at the Wall Street Corral : An Empirical Investigation of Error Types in Parser Output. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, p. 1048–1059, Jeju Island, Korea : Association for Computational Linguistics.

LE ROUX J., ROZENKNOP A. & CONSTANT M. (2014). Syntactic Parsing and Compound Recognition via Dual Decomposition : Application to French. In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics : Technical Papers*, p. 1875–1885, Dublin, Ireland : Dublin City University and Association for Computational Linguistics.

MOURET F. (2007). *Grammaire des constructions coordonnées. Coordinations simples et coordinations à redoublement en français contemporain.* PhD thesis, Université Paris 7 - Denis Diderot.

PETROV S., BARRETT L., THIBAUX R. & KLEIN D. (2006). Learning Accurate, Compact, and Interpretable Tree Annotation. In *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics*, p. 433–440, Sydney, Australia : Association for Computational Linguistics.

SAGOT B. (2010). The Lefff, a Freely Available and Large-coverage Morphological and Syntactic Lexicon for French. In *Proceedings of the Seventh conference on International Language Resources and Evaluation (LREC'10)*, Valletta, Malta : European Language Resources Association (ELRA).

SAGOT B. & BOULLIER P. (2005). From raw corpus to word lattices : robust pre-parsing processing. In *Proceedings of the 2nd International Conference Language And Technology (L&T'05)*, Poznań, Pologne.

SAGOT B. & DE LA CLERGERIE E. V. (2006). Error Mining in Parsing Results. In *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics*, p. 329–336, Sydney, Australia : Association for Computational Linguistics.

SEDDAH D., CANDITO M. & CRABBÉ B. (2009). Cross parser evaluation : a French Treebanks study. In *Proceedings of the 11th International Conference on Parsing Technologies (IWPT'09)*, p. 150–161, Paris, France : Association for Computational Linguistics.

SEDDAH D., TSARFATY R., KÜBLER S., CANDITO M., CHOI J. D., FARKAS R., FOSTER J., GOENAGA I., GOJENOLA GALLETEBEITIA K., GOLDBERG Y., GREEN S., HABASH N., KUHLMANN M., MAIER W., NIVRE J., PRZEPIÓRKOWSKI A., ROTH R., SEEKER W., VERSLEY Y., VINCZE V., WOLIŃSKI M., WRÓBLEWSKA A. & DE LA CLERGERIE E. V. (2013). Overview of the SPMRL 2013 Shared Task : A Cross-Framework Evaluation of Parsing Morphologically Rich Languages. In *Proceedings of the Fourth Workshop on Statistical Parsing of Morphologically-Rich Languages*, p. 146–182, Seattle, Washington, USA : Association for Computational Linguistics.

SIGOGNE A., CONSTANT M. & LAPORTE E. (2011). French parsing enhanced with a word clustering method based on a syntactic lexicon. In *Proceedings of the Second Workshop on Statistical Parsing of Morphologically Rich Languages*, p. 22–27, Dublin, Ireland : Association for Computational Linguistics.

THOMASSET F. & DE LA CLERGERIE E. V. (2005). Comment obtenir plus des métagrammaires. In *Proceedings of the Conference TALN 2005*, Dourdan, France : ATALA/AFCP.

WATANABE T. & SUMITA E. (2015). Transition-based Neural Constituent Parsing. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1 : Long Papers)*, p. 1169–1179, Beijing, China : Association for Computational Linguistics.

Interface syntaxe-sémantique au moyen d'une grammaire d'arbres adjoints pour l'étiquetage sémantique de l'arabe

Cherifa Ben Khelil[1, 2] Chiraz Ben Othmane Zribi[1] Denys Duchier[2] Yannick Parmentier[3]

(1) RIADI, Campus universitaire de la Manouba, 2010, Tunisie
(2) LIFO, Bâtiment IIIA 6 rue Léonard de Vinci, F-45067 Orléans, France
(3) LORIA - Projet SYNALP, Campus Scientifique, BP 23954 506 Vandoeuvre-les-Nancy CEDEX, France
cherifa.bk@gmail.com, chiraz.zribi@ensi-uma.tn, denys.duchier@univ-orleans.fr, yannick.parmentier@loria.fr

RESUME

Dans une grammaire formelle, le lien entre l'information sémantique et sa structure syntaxique correspondante peut être établi en utilisant une interface syntaxe/sémantique qui permettra la construction du sens de la phrase. L'étiquetage de rôles sémantiques aide à réaliser cette tâche en associant automatiquement des rôles sémantiques à chaque argument du prédicat d'une phrase. Dans ce papier, nous présentons une nouvelle approche qui permet la construction d'une telle interface pour une grammaire d'arbres adjoints de l'arabe. Cette grammaire a été générée semi automatiquement à partir d'une méta-grammaire. Nous détaillons le processus d'interfaçage entre le niveau syntaxique et le niveau sémantique moyennant la sémantique des cadres et comment avons-nous procédé à l'étiquetage de rôles sémantiques en utilisant la ressource lexicale ArabicVerbNet.

ABSTRACT

Syntax-semantic interface using Tree-adjoining grammar for Arabic semantic labeling.
In formal grammar, the link between semantic information and its corresponding syntactic structure can be established using a syntax/semantic interface that allows the construction of sentence meaning. Semantic role labeling helps to achieve this task by automatically associating semantic roles with each argument of the predicate of a sentence. In this paper, we present a new approach that allows the construction of such interface for a Tree adjoining grammar for Arabic. This grammar was generated semi automatically from a meta-grammar. We detail the process of interfacing between syntactic and semantic levels through semantic frames and how we proceeded to the semantic roles labeling using the lexical resource ArabicVerbNet.

MOTS-CLES : Grammaire d'arbres adjoints ; méta-grammaire ; interface syntaxe/sémantique ; étiquetage de rôles sémantiques ; cadre sémantique ; langue arabe.

KEYWORDS: Tree adjoining grammar; meta-grammar; syntax/semantic interface; semantic role labeling; semantic frame; Arabic language.

1 Introduction

La construction automatique du sens d'une phrase représente un grand intérêt pour le domaine du Traitement Automatique du Langage Naturel (TALN). Mais pour ce faire, il est souvent utile de

faire correspondre aux composantes syntaxiques de la phrase des représentations sémantiques. L'étiquetage de rôles sémantiques permet de réaliser cette tâche en associant automatiquement des rôles sémantiques à chaque argument du prédicat (par exemple un verbe) d'une phrase. Ces rôles expriment des rôles abstraits que les arguments d'un prédicat peuvent admettre dans un événement ainsi que leur relation probable avec la fonction syntaxique dans cette phrase. L'étiquetage de rôles sémantiques est utile pour diverses applications du domaine TALN tels que les systèmes de traduction automatiques (Liu & Gildea, 2010), les systèmes questions-réponses (Pizzato & Mollá, 2008 ; Maqsud et al., 2014) ou encore les systèmes d'extraction de l'information (Christensen et al., 2010 ; Fader et al., 2011). Plusieurs de ces approches utilisent les ressources PropBank (Kingsbury & Palmer, 2003) et FrameNet (Baker et al., 1998) afin de définir le prédicat, les rôles utilisés lors de l'étiquetage ainsi que l'ensemble de test pour l'apprentissage automatique. En ce qui concerne l'arabe nous pouvons citer les travaux de (Diab et al, 2008) qui utilisent les machines vectorielles (Vapnik, 1998) et (Meguehout et al, 2017) qui se sont basés sur le raisonnement à partir de cas pour réaliser l'étiquetage sémantique.

Dans une grammaire formelle, ce lien entre la sémantique et la syntaxe peut être établi en utilisant une interface syntaxe/sémantique. Cette dernière permet de superviser la construction du sens de la phrase en unifiant les informations sémantiques de ses constituants. Les représentations sémantiques peuvent être sous la forme d'une formule logique des prédicats (Joshi & Vijay-Shanker, 1999 ; Kallmeyer & Romero, 2004 ; Romero & Kallmeyer, 2005), une formule logique sous-spécifiée (Gardent & Kallmeyer, 2003 ; Parmentier, 2007) ou plus récemment sous la forme d'un cadre sémantique (Kallmeyer & Osswald, 2013). A notre connaissance, de tels travaux n'ont pas été menés sur l'arabe.

C'est dans ce contexte que s'inscrit notre travail de recherche qui vise à élaborer une grammaire d'arbres adjoints (TAG) (Joshi et al., 1975) décrivant la syntaxe et la sémantique de l'arabe standard moderne (ASM) en vue d'une analyse syntaxico-sémantique. La grammaire que nous proposons a été produite semi automatiquement grâce au langage de description méta-grammatical XMG (eXtensible MetaGrammar) (Crabbé et al, 2013). À partir de la description méta-grammaticale ArabicXMG nous avons généré ArabTAG V2.0 (Ben Khelil et al, 2016). Ensuite, nous avons étendu cette grammaire en intégrant des informations sémantiques. Notre choix s'est porté sur la sémantique des cadres. Ce choix est motivé par la facilité de l'interfaçage entre le niveau syntaxique et le niveau sémantique.

Cet article est organisé de la manière suivante. Dans la section 2, nous détaillons le processus d'intégration de la dimension sémantique dans la méta-grammaire. Ensuite, nous présentons les étapes effectuées pour l'étiquetage de rôles sémantiques. Finalement, dans la section 4, nous exposons les premiers résultats de l'évaluation de cet étiquetage.

2 Intégration de la dimension sémantique dans la méta-grammaire

ArabTAG V2.0 (Ben Khelil et al, 2016) a été générée à partir d'une description méta-grammaticale en utilisant le compilateur XMG2 (Petitjean, 2014). Elle couvre les phrases verbales (forme active et passive), les phrases nominales, les différents types des syntagmes nominaux et les syntagmes prépositionnels. Elle traite aussi les différents phénomènes linguistiques arabes tels que la variation des positions des éléments au sein des composants syntaxiques, les compléments supplémentaires, les règles d'accord et les formes agglutinées. Afin d'étendre notre méta-grammaire et produire une grammaire TAG à portée sémantique, nous avons pensé à associer aux familles des arbres décrites des cadres sémantiques (FillMore, 1982). Nous

nous sommes basés sur la théorie du linking[1] (Levin, 1993 ; Kasper, 2008). Selon cette théorie, le verbe permet d'exprimer dans la plupart des cas la sémantique d'un évènement ainsi que la relation entre ses participants. En effet, l'ensemble des rôles sélectionnés par un prédicat verbal constitue un cadre sémantique. Certains de ces rôles sont obligatoires et déterminent la présence ou non de certaines fonctions grammaticales. Par exemple, lorsque l'acteur du verbe est présent dans une phrase, il est en position sujet (au cas nominatif). Ce genre de composant peut avoir le rôle d'«Agent». Ainsi, la fonction grammaticale permet d'indiquer le rôle à attribuer.

Notre idée consiste à préciser les rôles sémantiques au niveau du prédicat, qui est le verbe, au sein des structures syntaxiques décrites dans notre méta-grammaire. Le cadre de la phrase est ensuite construit au fur et à mesure de l'analyse syntaxique en unifiant les cadres sémantiques élémentaires de ses composants syntaxiques par l'intermédiaire d'une interface syntaxe/sémantique.

2.1 Construction de l'interface syntaxe/sémantique

L'interface syntaxe-sémantique au niveau de notre méta-grammaire est effectuée de la manière suivante (voir figure 1) :

- Au niveau syntaxique des familles de classes décrites par la méta-grammaire, nous avons défini les arguments du prédicat (verbe). Ces familles regroupent les arbres ancrés par un verbe et un nœud (de substitution) pour chaque argument du prédicat.
- Au niveau sémantique, nous avons défini les rôles sémantiques du cadre du prédicat. La dimension <frame> permet de décrire un cadre sémantique à l'aide de structures de traits typées.
- Le lien entre les rôles sémantiques et les constituants syntaxiques est établi à l'aide de l'interface syntaxe/sémantique en utilisant la dimension <iface>. Cette dernière correspond à la définition, pour chaque classe, d'une matrice de traits. Cette matrice permet d'associer un nom global (le trait) à une variable (la valeur du trait) ce qui permettra d'unifier les variables (suite à une opération de substitution ou d'adjonction) du même nom global et faire la correspondance entre les arguments du prédicat et leurs rôles correspondants.
- Les cadres sémantiques élémentaires sont définis aux niveaux du lexique (les lemmes).

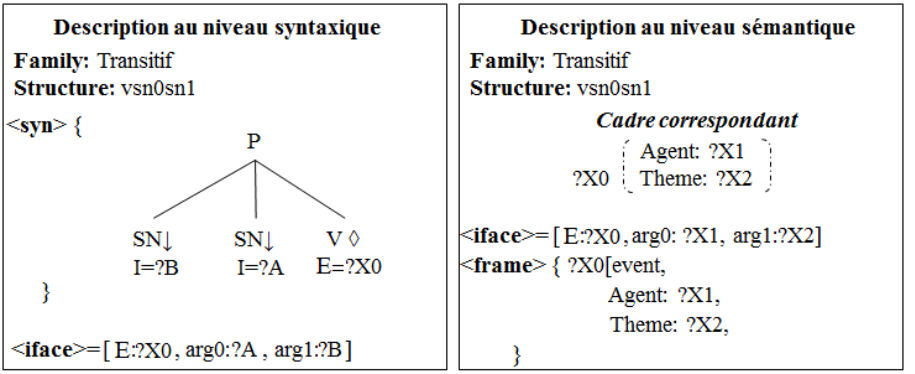


FIGURE 1 : Description de l'interface syntaxe/sémantique dans la méta-grammaire

[1] C'est la mise en relation d'une structure en rôle sémantique avec une structure syntaxique.

2.2 MAPPING entre la méta-grammaire et ArabicVerbNet

L'alimentation automatique de notre méta-grammaire par les rôles sémantiques se fait par l'intermédiaire de la ressource lexicale ArabicVerbNet (Mousser, 2010), version arabe de VerbNet (Kipper, 2008). Celle-ci couvre les verbes les plus utilisés de l'arabe standard moderne[2]. VerbNet est une ressource lexicale pour les verbes anglais qui repose sur le système de classification sémantico-syntaxique des verbes de (Levin, 1993). Les verbes ayant un comportement syntaxique et sémantique similaire sont affectés au même groupe de classes. Chaque classe d'un verbe est décrite au moyen des éléments suivants :

- Les membres : la liste des verbes appartenant à cette classe ou à une sous-classe. Cette liste contient aussi des informations sur la racine verbale, la forme déverbale et le participe de ces verbes.
- Les rôles : ce sont les rôles thématiques attribués à chaque membre du verbe de la classe. Ces rôles peuvent admettre un ensemble de restrictions sur leurs natures (animation, location, etc.).
- Les cadres : ils définissent la correspondance entre les rôles sémantiques et les arguments syntaxiques. Cette correspondance est expliquée à l'aide d'un exemple. En effet, pour chaque exemple de phrase, sa structure syntaxique et les relations sémantiques entre les arguments du prédicat sont définis.

Nous avons parcouru toutes les classes d'ArabicVerbNet. Dans un premier temps, les informations ont été regroupées selon la structure syntaxique de la phrase. Le groupement de ces structures respecte les familles d'arbres élémentaires définies par notre grammaire. Ensuite, pour chaque structure syntaxique, nous avons extrait l'ensemble des combinaisons des rôles sémantiques possibles pour construire les cadres sémantiques correspondants. Cet ensemble de cadre sémantique est défini au niveau de la méta-grammaire (avec la dimension <frame>). Au final, le compilateur XMG2 (Petitjean, 2014) compile cette méta-grammaire et génère une grammaire constituée d'un ensemble d'arbres élémentaires associés à leurs cadres prédicats.

3 Étiquetage à base de rôles sémantiques

L'affectation des cadres sémantiques se fait lors de l'analyse syntaxique par l'intermédiaire de l'interface syntaxe/sémantique. Au fur et à mesure qu'une phrase est analysée, son cadre sémantique est construit en unifiant les cadres sémantiques élémentaires de ses constituants et celui du verbe prédicat.

Prenons l'exemple de la phrase suivante (voir figure 2) : طاردَ الشرطيُ اللصَ (*le policier poursuit le voleur*). Le processus de construction du sens de cette phrase est réalisé comme suit : L'arbre syntaxique de la phrase analysée est constitué d'un verbe suivi d'un sujet et d'un objet. Le verbe ancré «طاردَ» (*poursuivre*) admet donc deux arguments. Après avoir effectué l'étiquetage de rôles sémantiques (en consultant ArabicVerbNet), les deux rôles attribués à ces arguments sont : Agent et Theme. Les cadres élémentaires (personnage et profession) sont associés aux arbres élémentaires des syntagmes nominaux. L'élément I qui représente l'interface syntaxe/sémantique permet le

[2] La version actuelle d'ArabicVerbNet comporte 334 classes qui contiennent 7672 verbes et 1393 cadres.

partage des variables de traits des nœuds avec les variables issues des cadres sémantiques. Les opérations de substitution déclenchent les équations d'unification entre ces variables : [X1 = A] et [X2 = B]. L'unification est ainsi opérée et mène à l'insertion des cadres élémentaires de «الشرطيُ» (*le policier*) et «اللصَ» (*le voleur*) dans le cadre sémantique prédicat du verbe «طاردَ» (*poursuivre*). Le cadre final obtenu représente le sens de la phrase.

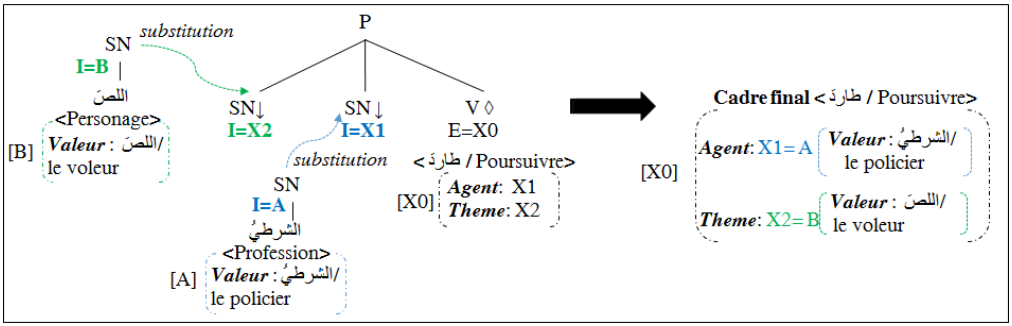


FIGURE 2: Composition du cadre sémantique pour طاردَ الشرطيُ اللصَ / *le policier poursuit le voleur*

Une structure syntaxique peut avoir plusieurs cadres sémantiques correspondants. Ces cadres représentent différents sens susceptibles d'occasionner plusieurs interprétations possibles. Par exemple : Un sujet peut être «Agent» ou «Acteur» selon les contextes. Nous avons exploité d'avantage les classes d'ArabicVerbNet et nous avons établi un ensemble de contraintes afin d'optimiser la tache de l'étiquetage de rôles sémantiques au moment de l'analyse sémantique :

- ***La classe du verbe :*** les cadres sémantiques pour un verbe sont définis en fonction sa classe.
- ***Le type de la préposition pour les syntagmes prépositionnels :*** certains rôles sémantiques ont tendance à apparaître comme des syntagmes prépositionnels. Dans ce cas, la préposition peut indiquer le sens de ce syntagme et ainsi intervenir pour restreindre le choix du cadre correspondant. Considérons l'exemple des deux phrases suivantes :
  (1) نبحَ الكلبُ على الهرّ/ *Le chien aboie sur le chat*
  (2) نبحَ الكلبُ منَ الخوفِ/ *Le chien aboie de peur*
  Selon la classe verbale de «نبح» (*aboyer*), animal_sounds-1 définie dans ArabicVerbNet, nous avons trois combinaisons possibles de cadres sémantiques pour la structure de ces deux phrases :
  a) Agent+ {prep (على)} +Recipient : la préposition «على» (*sur*) indique que le rôle sémantique de l'objet est Recipient.
  b) Agent+ {prep (من)} +Cause : la préposition «من» (*de*) exige que le rôle sémantique de l'objet soit Cause.
  c) Location+ {prep (ب)} +Agent : la préposition «بِ» (*avec*) indique que le rôle sémantique de l'objet est Agent.
  Après avoir filtrer ces résultats en tenant compte de la contrainte sur la préposition, nous obtenons les correspondances sémantiques suivantes :
  1) -a) Agent {الكلبُ/ *le chien*} + {prep (على/ *sur*)} +Recipient {الهرّ/ *le chat*}.
  2) -b) Agent {الكلبُ/ *le chien*} + {prep (من/ *de*)} +Cause {الخوفِ/ *peur*}.
- ***Les contraintes :*** un verbe peut imposer un ensemble de restrictions à ses rôles d'argument. Par exemple, en exigeant qu'un rôle soit humain et /ou animé, etc. Soient les deux phrases suivantes avec leurs interprétations sémantiques :
  (3) يحبُ عليٌ فاطمةً / *Ali aime Fatima*: Experiencer {عليٌ/ *Ali*} +Theme {فاطمةً/ *Fatima*}.

(4) يحبُ الكتابُ فاطمةً / *Le livre aime Fatima*: Experiencer {الكتابُ/ *le livre*} + Theme {فاطمةً/ *Fatima*}.

Le prédicat est le verbe «أحب» (*aimer*). Bien que les deux phrases soient syntaxiquement correctes la deuxième est sémantiquement incorrecte. Le sujet «الكتابُ» (*livre*) ne peut pas éprouver des sentiments envers un humain. Lors de l'analyse sémantique, nous faisons intervenir les contraintes spécifiées pour les rôles sémantiques au niveau de la classe du verbe «أحب» (*aimer*). Après avoir examiné cette classe, nous avons remarqué que l' «Experiencer» doit être animé et humain. Par conséquent, nous pouvons confirmer que la première phrase est sémantiquement correcte alors que la deuxième ne l'est pas, puisque son sujet ne satisfait pas cette contrainte.

4 Expérimentations

Afin d'évaluer notre grammaire dans sa tâche d'analyse syntaxico-sémantique, nous avons défini un corpus de test de 500 phrases (347 phrases verbales et 153 phrases nominales) extraites à partir d'un livre scolaire tunisien (niveau 8ième année[3]). Ce choix est dû à l'indisponibilité des corpus annotés d'information syntaxico-sémantique pour l'arabe standard moderne (Ben Khelil, 2017). Nous avons développé un outil afin d'effectuer cette analyse. Cet outil permet dans un premier lieu de faire un étiquetage morphosyntaxique des éléments de la phrase, suivit par l'analyse syntaxico-sémantique suivant les étapes expliquées dans la section 3.

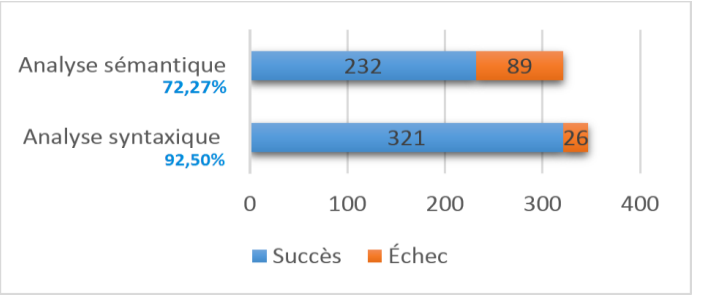


FIGURE 3 : Résultat de l'analyse syntaxico-sémantique des phrases verbales

Parmi les 347 phrases verbales testées, nous avons réussi à analyser syntaxiquement 321 phrases (92,50%) et sémantiquement 232 phrases (72,27%). Nous nous focalisons, dans cet article, à décrire l'évaluation de la partie sémantique. Les causes de l'échec de l'analyse sémantique sont principalement dues à un manque de couverture au niveau d'ArabicVerbNet. Nous avons constaté que 19,93% des verbes des phrases testées ne sont pas définis dans cette ressource. De plus, pour un verbe donné dans ArabicVerbNet, la liste des structures syntaxiques des phrases qu'il peut admettre n'est pas exhaustive. En effet, nous avons mesuré un taux de 5,29% d'échec d'analyse du à l'absence de la structure de la phrase correspondante au moment de l'étiquetage de rôles sémantiques. Nous avons aussi obtenu 2,49% d'échec pour les phrases complexes. L'analyse de ce genre de phrases est plus compliquée vu qu'elles contiennent plusieurs verbes.

[3] Equivalent à la 4ième année au collège en France.

5 Conclusion

Nous avons présenté une nouvelle approche visant à construire une grammaire d'arbres adjoints pour représenter la syntaxe et la sémantique de l'arabe. Nous nous sommes concentrés dans cet article sur le processus d'intégration de l'information sémantique. Notre idée est d'associer aux familles d'arbres élémentaires de la grammaire une sémantique à base de cadres et d'intégrer les rôles sémantiques à partir de la ressource ArabicVerbNet. Ceci a permis d'établir une correspondance entre arguments sémantiques et arguments syntaxiques par l'intermédiaire d'une interface syntaxe/sémantique permettant aux cadres sémantiques élémentaires de s'unifier lors de la composition syntaxique.

Lors de l'étiquetage de rôles sémantiques nous avons constaté que plusieurs informations peuvent aider à lever l'ambigüité sémantique. Nous citons ; la classe du verbe, les propriétés du rôle et aussi l'utilisation de certaines prépositions pour les syntagmes prépositionnels. Bien que les premiers résultats de l'analyse soient encourageants, nous envisageons dans le futur proche d'augmenter notre corpus de test et d'avoir recours à l'apprentissage automatique pour améliorer le taux de réussite et pallier le manque de données au niveau d'ArabicVerbNet.

Références

JOSHI A., LEVY L., TAKAHASHI M. (1975). Tree adjunct grammars. Journal of Computer and System Sciences. 10(1), 136 – 163.

CRABBÉ C., DUCHIER D., GARDENT C., LE ROUX J., PARMENTIER Y. (2013). XMG : eXtensible MetaGrammar. Computational Linguistics. 39(3), 591–629.

JOSHI A., VIJAY SHANKER K. (1999). Compositional Semantics with Lexicalized Tree Adjoining Grammar (LTAG) : How Much Underspecification is Necessary . In Proceedings of the Third International Workshop on Computational Semantics, IWCS-03, Tilburg, The Netherlands.

KALLMEYER L., ROMERO M. (2004). LTAG Semantics with Semantic Unification. In Proceedings of TAG+7, Vancouver, pages 155–162.

ROMERO M., KALLMEYER L. (2005). Scope and Situation Binding in LTAG using Semantic Unification. In Proceedings of the Sixth International Workshop on Computational Semantics IWCS-6, Tilburg.

KALLMEYER L., JOSHI A. (2003). Factoring Predicate Argument and Scope Semantics: Underspecified Semantics with LTAG. Research on Language and Computation, volume 1 :1-2, pages 3–58.

GARDENT C., KALLMEYER L. (2003). Semantic construction in FTAG. In Proceedings of the European chapter of the Association for Computational Linguistics (EACL'03), Budapest.

PARMENTIER Y. (2007). SemTAG : une plate-forme pour le calcul sémantique à partir de Grammaires d'Arbres Adjoints. Ph.D thesis, université Henri Poincaré – Nancy 1.

KALLMEYER L., OSSWALD R. (2013). Syntax-Driven Semantic Frame Composition in Lexicalized Tree Adjoining Grammars. Journal of Language Modelling Vol i2, pp. 1–63.

KINGSBURY P., PALMER M. (2003). Propbank: the next level of treebank. In Proceedings of Treebanks and Lexical Theories.

BAKER F., FILLMORE J., LOWE B. (1998). The berkeley FrameNet project. In COLINGACL '98: University of Montréal.

LIU D., GILDEA D. (2010). Semantic role features for machine translation. In Proceedings of the 23rd International Conference on Computational Linguistics (Coling 2010), pages 716–724, Beijing, China.

PIZZATO L.A., MOLLÁ D. (2008). Indexing on semantic roles for question answering. In Coling 2008: Proceedings of the 2nd workshop on Information Retrieval for Question Answering, pages 74–81, Manchester, UK, August.

MAQSUD U., ARNOLD S., HÜLFENHAUS M., AKBIK A. (2014). Nerdle: Topic-specific question answering using wikia seeds. In COLING (Demos), pages 81–85.

CHRISTENSEN J., MAUSAM ., SODERLAND S., ETZIONI O. (2010). Semantic role labeling for open information extraction. In Proceedings of the NAACL HLT 2010 First International Workshop on Formalisms and Methodology for Learning by Reading, pages 52–60, Los Angeles, California, June. Association for Computational Linguistics.

FADER A., SODERLAND S., ETZIONI O. (2011). Identifying relations for open information extraction. In Proceedings of the Conference on Empirical Methods in Natural Language Processing, pages 1535–1545. Association for Computational Linguistics.

DIAD M., MOSCHITTI A., PIGHIN D. (2008). Semantic role labeling systems for Arabic language using kernel methods. In: 46th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies (ACL 2008: HLT), Columbus, Ohio, USAS.

MEGUEHOUT H., BOUHADADA T., LASKRI M.T. (2017). Semantic role labeling for Arabic language using case-based reasoning approach. Int J Speech Technol (2017) 20: 363. https://doi.org/10.1007/s10772-017-9412-6

VAPNIK N. (1998). Statistical Learning Theory. JohnWiley and Sons.

LEVIN B. (1993). English Verb Classes and Alternations A Preliminary Investigation. Chicago: University of Chicago Press.

KASPER S. (2008). A comparison of thematic role theories," Master Homework. Marburg University.

KIPPER K., KORHONEN A., RYANT N., PALMER M. (2008). A large-scale classification of English verbs Lang. Resour. Eval. J., 42 (2008), pp. 21-40.

MOUSSER J. (2010). A large coverage verb taxonomy for Arabic. In: Seventh Conference on International Language Resources and Evaluation (LREC'10), Valetta, Malta, pp. 2675–2681.

BEN KHELIL C. (2017). Générer une grammaire d'arbres adjoints pour l'arabe à partir d'une méta-grammaire. Présenté sous forme de poster à la 24° édition de la conférence TALN aux 19es Rencontres des Étudiants Chercheurs en Informatique pour le Traitement Automatique des Langues (RÉCITAL 2017) Orléans, France.

BEN KHELIL C., DUCHIER D., PARMENTIER Y., ZRIBI C., BEN FRAJ F. (2016). ArabTAG : from a Handcrafted to a Semi-automatically Generated TAG. In TAG+12: 12th International Workshop on Tree-Adjoining Grammars and Related Formalisms, Düsseldorf, Germany.

PETITJEAN S. (2014). Génération Modulaire de Grammaires Formelles. Ph.D. thesis, Université d'Orléans, France.

FinSentiA: Sentiment Analysis in English Financial Microblogs

Thomas Gaillat (1), Annanda Sousa (1), Manel Zarrouk (1), Brian Davis (2)
(1) Insight Centre for Data Analytics, IDA Business Park, Ireland
(2) Maynooth University, Ireland
firstname.surname@insight-centre.org, brian.davis@mu.ie

RÉSUMÉ

FinSentiA: Analyse de Sentiments dans les Microblogs Financiers en Anglais

L'objectif de cet article est de présenter la construction d'un système d'analyse de sentiments dans le domaine des microblogs financiers en anglais. Le but de notre travail est de construire un classifieur pour la prédiction de sentiments chez les investisseurs financiers sur les plateformes de microblogs telles que StockTwits et Twitter. Notre contribution montre qu'il est possible de mener une analyse fine des sentiments. Après extraction des entités financières et leurs contextes, le système attribue des scores en valeurs continues. Il repose sur une approche par réseaux profonds pour la méthode de classification. Les résultats montrent un F1-score de 0.85 (2 classes) et une valeur de similarité cosinus de 0.62.

ABSTRACT

FinSentiA: Sentiment Analysis in English Financial Microblogs

The objective of this paper is to report on the building of a Sentiment Analysis (SA) system dedicated to financial microblogs in English. The purpose of our work is to build a financial classifier that predicts the sentiment of stock investors in microblog platforms such as StockTwits and Twitter. Our contribution shows that it is possible to conduct such tasks in order to provide fine-grained SA of financial microblogs. We extracted financial entities with relevant contexts and assigned scores on a continuous scale by adopting a deep learning method for the classification. Results show a 0.85 F1-Score on a two-class basis and a 0.62 cosine similarity score.

MOTS-CLÉS : Analyse de Sentiments, Entités Financières, Valeurs Continues, Fouille de Données d'Opinions, Granularité Entité

KEYWORDS: Sentiment Analysis, Financial Entities, Continuous Scale, Opinion Mining, Entity level

1 Introduction

Stock investment platforms like StockTwits[1] provide their users with many indicators of live events occurring on markets. To anticipate volatility, seen as a risk, investors use indexes such as the VIX[2] (Volatility IndeX) which is interpreted as a measure of investor confidence with respect to S&P 500 stock index option prices (Blitzer, 2017). The VIX is calculated from numerical values which module investor sentiment but there is an increasing interest in capturing opinions on finance from social media as an alternative source of sentiment. In a highly competitive and volatile domain such as investing, acquiring an insight into the public opinion of relevant and valuable economic signals can give a competitive edge and allow more informed investment decisions to be executed. Microblog messages posted on social media such as Twitter or StockTwits are central to determining these economic signals.

Sentiment Analysis (SA) plays a central role for this purpose. Intersecting between the fields of Computational Linguistics and Natural Language Processing, the field has a long tradition in the use of tools to automatically determine sentiments in documents (Liu, 2012; Pang & Lee, 2008). It is admitted that SA tends to perform best when restricted to a specific domain and much work, so far, has focused on domains such as hotel and product reviews. In the financial domain some studies have used press articles with economic and financial focuses (Malo et al., 2013a; Malo et al., 2013b) but few studies were conducted on financial microblog messages (Bollen et al., 2011). And yet, due to the specific style of this type of messages, financial microblog SA requires adapted NLP methods. In addition, despite the need of financial experts to have a more revealing fine-grained analysis, most studies so far have concentrated on classifying sentiments in categories at sentence or document level. Recently, there have been some studies focusing on entities on a continuous scale (Cortis et al., 2017), but they were based on already-identified text spans which simplified the sentiment assignment task. Finally, many studies have used supervised learning approaches but, as far as we know, few have applied deep learning methods.

The purpose of our research is to develop a fine-grained financial SA classifier for the SSIX[3] project. In this paper, we present a supervised learning approach[4] which relies on a financial microblog Gold Standard for the training of a neural network. The SA is carried out in a fine-grained fashion: i) by extracting entities, i.e. stocks, along with relevant contexts ii) by assigning sentiment scores on a continuous polarity scale ranging from -1 to 1. The remaining of this paper is divided into four sections. In Section 2, we provide a short review of related work in the domain of financial SA. Section 3 covers the method used to build the financial classifier. In Section 4, we present the results followed by a short discussion. We conclude in Section 5.

[1] See https://stocktwits.com

[2] Cboe created the Volatility Index® (VIX® Index) which is a benchmark index to measure the market's expectation of future volatility.

[3] Social Sentiment IndeX is a sentiment analysis platform dedicated to financial microblogs

[4] This financial classifier is operational and available for users as it is implemented in the SSIX online platform. See https://ssix-project.eu/knowledgebase_category/demos/

2 Related work in financial sentiment analysis

Given the broad diversity of studies conducted in the domain of SA to date (see Liu, 2012; Pang & Lee, 2008 for a detailed overview of the domain) and due to the scope of the SSIX project, we choose to focus this review of related work to the financial domain only. There have been two types of approaches: rule-based and supervised learning approaches. Rule-based approaches consist in dictionary lookups for the assignment of polarities. Typically, the first step of the task consists in building a lexicon of terms from a domain-specific corpus together with their respective polarities (Loughran & McDonald, 2011; Moreno-Ortiz & Fernández-Cruz, 2015; Tetlock, 2007). The second step of the rule-based approach is to create rules for polarity lookups. (see (Loughran & McDonald, 2011; Malo et al., 2013a; Malo et al., 2013b; Tetlock, 2007; Wiebe et al. 2005) for variations in the approach).

The second type of approach in SA relies on supervised learning methods. When applied to financial texts, these Machine Learning (ML) methods make use of annotated documents that are used to "learn" specific financial features in order to subsequently classify. Some studies use common feature engineering strategies relying on internal text features such as bag-of-words, Part-of Speech (POS) tags and Named Entities (Antweiler & Frank, 2004; O'Hare et al., 2009; Schumaker et al., 2012; Sprenger et al. 2014). Other studies add polarities from lexicons as features of vectors (Bollen et al., 2011; Malo, et al., 2013b) and assign sentiment classes at sentence level. In terms of results, (Bollen et al., 2011) reported best results with a 0.869 three-class accuracy.

More recently, as part of the SSIX project (Davis et al., 2016), (Cortis et al., 2017) gave the opportunity for SemEval-2017 candidates to conduct experiments based on the data set used in our experiment. The best scores were obtained by combining features such as POS, Word embeddings (Mikolov et al., 2013) and financial lexicons in ML approaches including neural network architectures (Ghoshal et al., 2017; Jiang et al., 2017).

The presented approaches focus on SA at sentence/document level while our finer-grained approach targets entities on a continuous scale. We use a different implementation of cosine similarity that targets entities, and we do not use manually selected text spans for sentiment assignment.

3 Method

This section covers the method used to build the financial classifier. We describe the evaluation corpus before detailing the experimental setup.

3.1 Evaluation corpus

The corpus[5] is a data set that was made available for SemEval-2017 Task 5 (Cortis et al., 2017). It was specifically created for the purpose of financial SA in microblogs. It is a collection of 2,002 microblog messages in English from the Twitter and StockTwits platform. The messages were manually annotated to assign one sentiment score per financial entity (i.e. stocks) of each message.

[5] This data set was published as a Gold Standard and is publicly available from https://ssix-project.eu. For legal reasons, it only includes 1,336 StockTwits messages.

Table 1 shows the distribution of sentiments. To rate the sentiments on stocks—i.e, *cashtags* such as $AAPL for Apple—the scores were given on a continuous scale of [-1; 1] from *bearish*[6] to *bullish*[7]. To measure reliability, inter-rater agreement was calculated at entity level using Fleiss's Kappa and Krippendorff's alpha (0.69 and 0.61 respectively) prior to consolidating raters' scores. For more on this Gold Standard (GS) see (Gaillat, Zarrouk, Freitas, & Davis, 2018). For the experiment (Section 3.3), the corpus is divided into two subsets on a 80-20% basis. The training set accounts for 1,761 messages.

	Positive Sentiments	Negative Sentiments
Training set	1,931	934
Test set	393	197

TABLE 1: Distribution of positive and negative sentiments for entities in the training and test sets

3.2 Evaluation metrics

We use two evaluation metrics for the purpose of our study. The first and main evaluation metric, implemented in Semeval-2017 task 5 (Cortis et al., 2017), is the cosine similarity function which uses two vectors of values, i.e. the Gold Standard (GS) scores per entity and the predicted scores by the classifier. A score of 1 indicating perfect similarity. The second metric (not required in SemEval-2017) is provided as an additional measure. It is based on recoding continuous values into two positive and negative categories ([-1, 0[and [0, 1] intervals) the two possible polarities. We then conduct precision and recall as well as overall accuracy measurements per class. This evaluation method is implemented to provide a basis for comparison with other studies even though they may use a different number of classes. Calculating the cosine of the angle between the two vectors provides a measurement of the overall similarity between the two sets of values (Ghosh et al., 2015; Jurafsky & Martin, 2009). The metric gives results on a scale of [-1;1], with 1 indicating perfect match between the two lists of values (See Formula (1)). Let G be the vector of values from the evaluation corpus, and let P be the vector of predicted values.

$$(1) \quad cosine(G,P) = \frac{\sum_{i=0}^{n} (Gi.Pi)}{\sqrt{\sum_{i=0}^{n} Gi^2}.\sqrt{\sum_{i=0}^{n} Pi^2}}$$

3.3 Experimental set up

This section details the experimental set up involving linguistic preprocessing (message splitting and cleaning), feature engineering and deep learning classification.

6 *Bearish* indicates the belief that the price of an asset will fall.

7 *Bullish* is the opposite of *bearish*

3.3.1 *Linguistic preprocessing*

(Cortis et al., 2017) relied on the message spans that matched entities and scores, thus avoiding noisy data. However, in our case, we dealt with entire messages to mirror the reality of incoming data into the system. Consequently, it was necessary to apply a method in order to split sentences into segments that contain entities with relevant contexts. Example (1) shows that $IBM and $WYNN are target entities that are not adjacent and are separated contextually with different sentiments.

(1) "$IBM was THE play today *but* this $WYNN ain't bad either"

The splitting strategy is twofold. Firstly, an algorithm determines whether entities should be treated as one single target or independently on the basis of adjacency. Adjacent entities are treated as one and non-adjacent entities are treated separately. Secondly, another algorithm determines whether messages should be split and how. This algorithm uses the targets determined by the first algorithm. It essentially splits sentences if targets are separated by more than one word. As a result, the sentence in Example 1 is split in two segments at the *but* coordination conjunction. Sentences also undergo stemming along with stop-word and punctuation elimination in order to strip messages of noisy data. To create an abstract representation of each segment, feature engineering is applied next. The influence of the performance of this pre-processing was evaluated by conducting classification with and without it. Results with version 2.2 of the model show an improvement in cosine similarity scores from 0.57 to 0.62.

3.3.2 *Feature engineering*

At this point, the messages are converted into a machine-readable vector representation. Each split message goes through a vectorisation process in which words are turned into feature values. To build the vector space, we use different four types of features. Compared with (Jiang et al., 2017), their work presented a bigger selection of features (12 types), several of them are the same as those we use, e.g. Bag of Words and sentiment lexicons such as Afinn[8] and SentiWordNet[9]. We experimented with some features implemented in Jiang's model such as Word Embedding (Google W2V[10]), and more sentiment lexicons (Bing Liu opinion lexicon[11], General Inquirer lexicon[12], MPQA[13]). However, in our proposed model the best result of features combination is shown in Table 1. As regard (Ghosal et al., 2017), they only used features from Word Embedding models.

[8] Available at http://www2.imm.dtu.dk/pubdb/views/publication_details.php?id=6010
[9] Available at http://sentiwordnet.isti.cnr.it/
[10] Google word to vector at https://code.google.com/archive/p/word2vec/
[11] Available at https://www.cs.uic.edu/~liub/FBS/sentiment-analysis.html
[12] Available at http://www.wjh.harvard.edu/~inquirer/homecat.htm
[13] Available at http://mpqa.cs.pitt.edu/

Features	Description
Bag of words	After stemming, the vocabulary size is 1,006 dimensions. Feature binary values are embedded within a token representation of messages that keeps track of word order.
Sentiword Net lexicon	Feature values are extracted from SentiWordNet (Baccianella et al., 2010) in which words are assigned positive and negative polarity in a [0;1] interval. They are embedded in a token representation of each sentence to keep track of the order of sentiments in a sentence.
AFINN[14]	Feature values are extracted from a list of 2,477 words and phrases with polarity information ranging from -5 to 5 (Nielsen, 2011)
Vader predicted sentiment	Feature value provided by Vader (Hutto & Gilbert, 2014), a rule-based heuristic relying on lexicon and aggregation for scoring.

TABLE 2: Lexicon and linguistic features used for classification

3.3.3 Deep learning classification

The classification process is run through a neural network using a deep learning model with recurrent LSTM and dense layers. These layers have 5 and 7 hidden layers respectively. This number of layers is the result of a layer incremental process to maximise results. (Figure 1 shows the neural network structure.

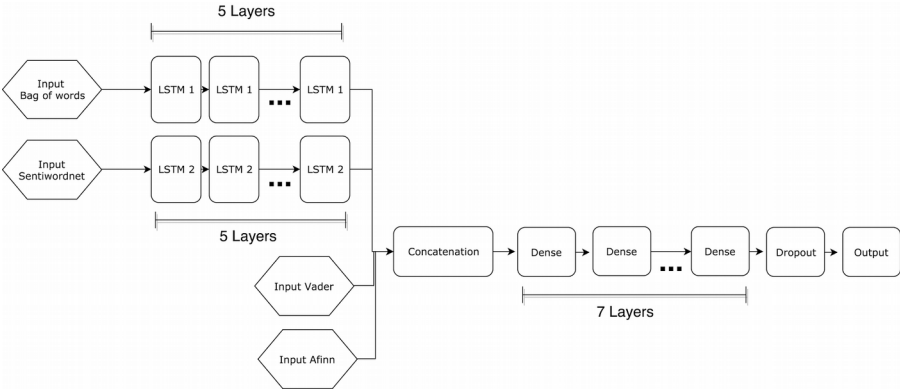


FIGURE 1: The structure of the neural network used in the financial classifier

The process is divided into two stages. Firstly, bag of words and Sentiwordnet (Baccianella et al., 2010) feature vectors are used as input for two parallel sets of five LSTM layers. These reduce the initial feature representation to optimize the number of dimensions by keeping the most significant features. The LSTM layers result in two vectors which are subsequently concatenated with the other feature vectors built with Vader (Hutto and Gilbert 2014) and Afinn (Nielsen, 2011). Secondly, the resulting concatenation is used as input for seven dense layers. A 25% dropout layer (Srivastava et al., 2014) is added because it is an efficient approach to prevent dataset overfitting. The output is a sentiment value included in the [-1;1] interval. Our neural network structure is similar to (Ghosal et

[14] Available at http://www2.imm.dtu.dk/pubdb/views/publication_details.php?id=6010

al., 2017) structure on the matter of having a set of LSTM with Dense layers. But their model differs from ours regarding the number (smaller) of layers and nodes from both the LSTM and Dense layers. In addition, they created an ensemble of neural networks to produce their final classifier model.

4 Results and discussion

Figure 2 shows the distribution of the predicted (right) and the test set (left) of sentiment scores at entity level. We can see clearly that the scores predicted by the classifier have the same distribution pattern as the ones from the test set from the Gold Standard. We notice, though, that there is a lack of predicted scores in the intervals [0.5, 1] and [-0.5, -1]. This can be explained by the inability of our model to learn from the training set due to the very small number of scores in these intervals in the GS. This can be seen in the scatter plot of the training set occurrences (Figure 3).

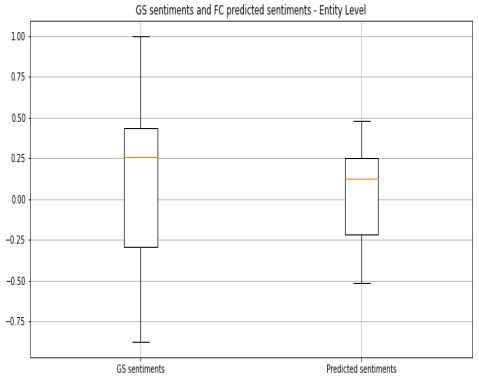


FIGURE 2: Distribution of the predicted and GS sets of sentiment scores at entity level

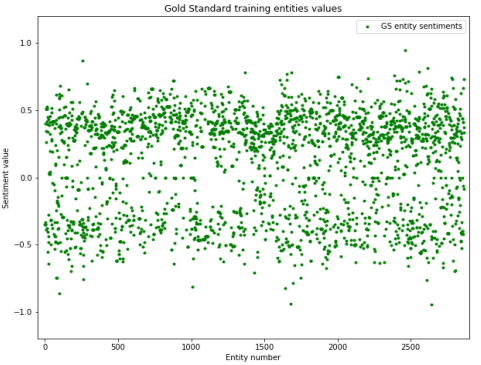


FIGURE 3: Dispersion of sentiment scores in the training set

As the closest existing work to ours is (Cortis et al. 2017), we computed our system performance using the exact same evaluation method they used. We decided not to use the predefined manually selected spans they use. Instead, we kept the real world configuration, which means dealing with entire messages, as they are, even if that implies adding a preprocessing step and dealing with the noise. Our system had 0.713 versus 0.733 for (Jiang et al. 2017).

As their evaluation method calculate the average of the cosine similarities at message level and not at entity level (both for the predicted and the test set), we decided to modify the method (see section 3.2) for it to take into consideration the entity scores distinctively, which keeps the fine-grained characteristic.

As we cannot compare totally our work to theirs we decided to have an internal baseline. We built a baseline model with Word2vec and SentiWordNet as features run through a 1-hidden-layer neural network. The cosine similarity function gives results taking into consideration the continuous values of the scores (see Table 3 for comparisons with the baseline model).

One point of discussion is about the cosine similarity measure which is the most used in the sentiment analysis domain. A more suitable evaluation metric that takes into consideration not only

the semantic similarity but also the sentiment similarity (Mohtarami et al. 2012) as not only the semantic space is important but also the emotional space.

To add another perspective to our results we computed as well the Accuracy and the F1-Scores of our model on a negative and positive classes basis ( ]-1,0] and]0,1]). Our main evaluation metric still the one described above (section 3.2) as it maintains the fine grained aspect of our work.

Entity Level	Financial Classifier version 1.0 (Baseline)	Financial Classifier version 2.2
Cosine Similarity	0.3352	**0.6269**
Accuracy	0.7801	0.8028
F1-Scores	0.8098	0.8522

TABLE 3: Accuracy and cosine similarity between predicted scores and Gold Standard results for the classification of financial entities in the Gold Standard

One point of discussion is about the cosine similarity measure. It may be relevant for semantic similarity but, in the case of sentiment similarity, it is not only the proximity of the vectors that is important but also the polarities of the values. Two vectors may be close to each other and yet have opposite values. This distinction is important in the measurement of sentiments.

5 Conclusion

Building a financial classifier for SA in the domain of financial microblogs faces many challenges involving linguistic engineering and machine learning tasks. Our contribution shows that it is possible to conduct such tasks in order to provide fine-grained SA of financial microblogs. We extracted financial entities with relevant contexts and assigned scores on a continuous scale by adopting a deep learning method for the classification. Further steps involve exploring a more suitable evaluation metric to our case, broadening the classifier features, increasing the training sets and applying some sensitivity analysis.

References

ANTWEILER, W., & FRANK, M. Z. (2004). Is All That Talk Just Noise? The Information Content of Internet Stock Message Boards. *The Journal of Finance*, *59*(3), 1259–1294.

BACCIANELLA, S., ESULI, A., & SEBASTIANI, F. (2010). SentiWordNet 3.0: An Enhanced Lexical Resource for Sentiment Analysis and Opinion Mining. Presented at the Proceedings of the Seventh Conference on International Language Resources and Evaluation (LREC'10), Valletta, Malta: European Language Resources Association (ELRA).

BLITZER, D. M. (2017). *S&P 500® - S&P Dow Jones Indices - Index Methodology*. New York, USA: S&P Dow Jones Indices.

BOLLEN, J., MAO, H., & ZENG, X.-J. (2011). Twitter mood predicts the stock market. *Journal of Computational Science*, *2*(1), 1–8.

CORTIS, K., FREITAS, A., DAUDERT, T., HUERLIMANN, M., ZARROUK, M., HANDSCHUH, S., & DAVIS, B. (2017). SemEval-2017 Task 5: Fine-Grained Sentiment Analysis on Financial Microblogs and News. In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)* (pp. 519–535). Vancouver, Canada: ACL.

DAVIS, B., CORTIS, K., VASILIU, L., KOUMPIS, A., MCDERMOTT, R., & HANDSCHUH, S. (2016). Social Sentiment Indices Powered by X-Scores. In *ALLDATA 2016 , The Second International Conference on Big Data, Small Data, Linked Data and Open Data*. Lisbon, Portugal: International Academy, Research, and Industry Association (IARIA).

GAILLAT, T., ZARROUK, M., FREITAS, A., & DAVIS, B. (2018). The SSIX Corpus: A Trilingual Gold Standard Corpus for Sentiment Analysis in Financial Microblogs. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*. Miyazaki, Japan: ELRA.

GHOSH, A., LI, G., VEALE, T., ROSSO, P., SHUTOVA, E., REYES, A., & BARNDEN, J. (2015). SemEval-2015 Task 11: Sentiment Analysis of Figurative Language in Twitter. In *Proceedings of the 9th International Workshop on Semantic Evaluation*. Denver, USA: Association for Computational Linguistics.

GHOSHAL, D., BHATNAGAR, S., EKBAL, A., SHAD AKHTAR, M., & BHATTACHARYYA, P. (2017). IITP at SemEval-2017 Task 5: An Ensemble of Deep Learning and Feature Based Models for Financial Sentiment Analysis. In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*. Vancouver, Canada: Association for Computational Linguistics.

HUTTO, C. J., & GILBERT, E. (2014). VADER: A Parsimonious Rule-Based Model for Sentiment Analysis of Social Media Text. In *Eighth International AAAI Conference on Weblogs and Social Media*. Ann Arbor, Michigan, USA: University of Michigan.

JIANG, M., LAN, M., & WU, Y. (2017). ECNU at SemEval-2017 Task 5: An Ensemble of Regression Algorithms with Effective Features for Fine-Grained Sentiment Analysis in Financial Domain. In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)* (pp. 888–893). Vancouver, Canada: Association for Computational Linguistics.

JURAFSKY, D., & MARTIN, J. H. (2009). *Speech and Language Processing (2nd Edition)*. Upper Saddle River, NJ, USA: Prentice-Hall, Inc.

LIU, B. (2012). *Sentiment Analysis and Opinion Mining*. San Rafael, Calif.: Morgan & Claypool Publishers.

LOUGHRAN, T., & MCDONALD, B. (2011). When is a Liability not a Liability? Textual Analysis, Dictionaries, and 10-Ks. *The Journal of Finance*, *1*(66), 35–66.

MALO, P., SINHA, A., TAKALA, P., AHLGREN, O., & LAPPALAINEN, I. (2013). Learning the Roles of Directional Expressions and Domain Concepts in Financial News Analysis. In *Proceedings of the 2013 IEEE 13th International Conference on Data Mining Workshops* (pp. 945–954). Washington, DC, USA: IEEE Computer Society.

MALO, P., SINHA, A., TAKALA, P., KORHONEN, P., & WALLENIUS, J. (2013). Good Debt or Bad Debt: Detecting Semantic Orientations in Economic Texts. *Journal of the Association for Information Science and Technology*, *65*(4), 782–796.

MOHTARAMI, MITRA, HADI AMIRI, MAN LAN, THANH PHU TRAN, AND CHEW LIM TAN. (2012). Sense Sentiment Similarity: An Analysis. In *Proceedings of the Twenty-Sixth AAAI Conference on Artificial Intelligence*, 1706–1712. AAAI'12. Toronto, Ontario, Canada: AAAI Press.

MORENO-ORTIZ, A., & FERNÁNDEZ-CRUZ, J. (2015). Identifying Polarity in Financial Texts for Sentiment Analysis: A Corpus-based Approach. *Procedia - Social and Behavioral Sciences*, *198*, 330–338.

NIELSEN, F. Å. (2011). A new ANEW: Evaluation of a word list for sentiment analysis in microblogs. In *Proceedings of the ESWC2011 Workshop on 'Making Sense of Microposts': Big things come in small packages* (Vol. 718, pp. 93–98). Heraklion, Greece: CEUR Workshop Proceedings.

O'HARE, N., DAVY, M., BERMINGHAM, A., FERGUSON, P., SHERIDAN, P., GURRIN, C., & SMEATON, A. F. (2009). Topic-dependent Sentiment Analysis of Financial Blogs. In *Proceedings of the 1st International CIKM Workshop on Topic-sentiment Analysis for Mass Opinion* (pp. 9–16). New York, USA: ACM.

PANG, B., & LEE, L. (2008). Opinion Mining and Sentiment Analysis. *Found. Trends Inf. Retr.*, *2*(1–2), 1–135.

SCHUMAKER, R. P., ZHANG, Y., HUANG, C.-N., & CHEN, H. (2012). Evaluating sentiment in financial news articles. *Decision Support Systems*, *53*(3), 458–464.

SPRENGER, T. O., TUMASJAN, A., SANDNER, P. G., & WELPE, I. M. (2014). Tweets and Trades: the Information Content of Stock Microblogs. *European Financial Management*, *20*(5), 926–957.

SRIVASTAVA, N., HINTON, G., KRIZHEVSKY, A., SUTSKEVER, I., & SALAKHUTDINOV, R. (2014). Dropout: A Simple Way to Prevent Neural Networks from Overfitting. *Journal of Machine Learning Research*, *15*, 1929–1958.

TETLOCK, P. C. (2007). Giving Content to Investor Sentiment: The Role of Media in the Stock Market. *The Journal of Finance*, *62*(3), 1139–1168.

WIEBE, J., WILSON, T., & CARDIE, C. (2005). Annotating Expressions of Opinions and Emotions in Language. *Language Resources and Evaluation*, *39*(2–3), 165–210.

L'optimisation des plongements de mots pour le français : une application de la classification des phrases

Jungyeul Park
CONJECTO, 74 rue de Paris, 35000 Rennes, France
http://www.conjecto.com

RÉSUMÉ

Nous proposons trois nouvelles méthodes pour construire et optimiser des plongements de mots pour le français. Nous utilisons les résultats de l'étiquetage morpho-syntaxique, de la détection des expressions multi-mots et de la lemmatisation pour un espace vectoriel continu. Pour l'évaluation, nous utilisons ces vecteurs sur une tâche de classification de phrases et les comparons avec le vecteur du système de base. Nous explorons également l'approche d'adaptation de domaine pour construire des vecteurs. Malgré un petit nombre de vocabulaires et la petite taille du corpus d'apprentissage, les vecteurs spécialisés par domaine obtiennent de meilleures performances que les vecteurs hors domaine.

ABSTRACT

Optimization of Word Embeddings for French : an Application of Sentence Classification.

We propose three novel methods for building word embeddings for French. We use results from part of speech tagging, detection of multiword expressions and lemmatization for a continuous vector space. For evaluation, we use these embedding vectors in a sentence classification task and compare them with the baseline embedding vector. We also explore domain adaptation approach for building embedding vectors, in which even with a small number of vocabularies and the small size of the training corpus, in-domain embeddings perform better than out-domain embeddings.

MOTS-CLÉS : Plongements de mots, catégorie grammaticale, expressions multi-mots, lemme, classification des phrases, français.

KEYWORDS: word embeddings, part of speech, multiword expression, lemma, sentence classification, French.

1 Introduction

Word embedding techniques have prevailed in natural language processing (NLP) and have obtained impressive results in several areas (Erhan *et al.*, 2010). Word embeddings are trained on word co-occurrence in text, and can capture semantic information about words and their meanings. Word2vec (Mikolov *et al.*, 2013) and GloVe (Pennington *et al.*, 2014) have been proposed to learn a distributed representation for words in a continuous vector space. Word embedding vectors with subwords were also presented to improve embedding quality (Luong *et al.*, 2013; Bojanowski *et al.*, 2017). There are still many *2vec-style variations. Multilingual embedding (Ammar *et al.*, 2016) or embedding with polysemy (Arora *et al.*, 2016) have also been presented as one extension of word embedding

techniques. At syntactic level, Levy & Goldberg (2014) proposed dependency-based word embeddings, and He *et al.* (2018) developed even further to encode the syntactic-aware context of entities. As proposed in previous work word embedding techniques have rapidly been used in a large range of applications in NLP over the last years.

In this paper, we present three methods for building and optimizing embedding vectors for French. We use POS tagging, multiword expression detection and lemmatization of words to optimize a word embedding task. We extrinsically evaluate the proposed method for embedding vectors by using a sentence classification task. The current work differs from previously proposed opinion mining (or sentiment analysis) (*inter alia*, Pang & Lee (2008)) where the classification task in previous work is mainly based on the document. For sentence classification, Socher *et al.* (2012) shows phrase fragments classification using a recursive neural network model. More recently, Dernoncourt *et al.* (2017a) proposes the sequential sentence classification task by adding a sentence label prediction layer.

The main contribution of this paper is as follows. First, we explore linguistically motivated methods to optimize word embeddings for French. Secondly, we propose a new data set for the sentence classification task for French. In addition to embedding optimization and sentence classification, we introduce a domain adaptation approach by selecting the training corpus for embeddings. We find several practical facts in word embeddings including the effects on the size of the training corpus, the size of vocabularies, and their relatedness with the task.

2 Optimizing Word Embeddings for French

This section describes three linguistically motivated methods for optimizing word embedding vectors for French : POS-aware, MWE-aware and lemma-aware embeddings. Instead of directly using surface forms of the word, we first use a pair of word and part of speech (POS) to disambiguate possible polysemy. Secondly, since words in multiword expressions (MWEs) can give more senses when they are bonded together, MWEs are dealt as a single unit. Thirdly, we use lemma forms in order to avoid sparsity because of rarely used inflected words. While the current methods have been proposed and used in the bag-of-words model, to our best knowledge there are no previous efforts for word embeddings to optimize the vector space model for French. We note that, while Ferré (2017) proposed complex terms for embeddings, he worked on embeddings for English.

Previously, character (Chrupała, 2013; Chen *et al.*, 2015; Wieting *et al.*, 2016) syllable (Yu *et al.*, 2017; Choi *et al.*, 2017), and subword (Mikolov *et al.*, 2012; Bojanowski *et al.*, 2017; Pinter *et al.*, 2017; González-Gallardo & Torres-Moreno, 2017; Stratos, 2017) embedding techniques have been proposed to enrich word vectors.

2.1 POS-aware embedding

POS tagging is one of the simplest, but the most important and well-studied tasks in NLP. Various supervised and unsupervised approaches have been proposed for POS tagging. Besides traditional rule-based approaches, in which POS dictionaries and manually crafted rules (*e.g.* syntagmatic patterns) are required, there are several supervised learning methods that can learn from POS tagged data such as transformation-based learning (Brill, 1995), hidden Markov models (HMM) (Kupiec,

... l'/O *arbitraire*/O *de*/O *la*/O *démesure*/O *veut*/O *susciter*/O *à*/B-MWE *tout*/I-MWE *prix*/I-MWE ./O

FIGURE 1 – MWE-annotated corpus from the French treebank

1992), maximum entropy models (Ratnaparkhi, 1996), Conditional random fields (CRF) (Lafferty *et al.*, 2001), etc. POS labels can give an additional information source for the word, especially some cases for homonyms and homographs. For example, while a noun *joue* ('cheek') and a verb *joue* ('play') would be considered as the same word in the general word embedding, *joue*/N and *joue*/V can be distinguishable. Therefore, we use a pair of the word and its POS label as a single unit for embeddings. Training a model with POS labels has also been proposed in statistical machine translation as one of factored training.

Any sequence labeling algorithm easily achieves state of arts results for POS tagging. We train and evaluate POS tagging using TnT (Brants, 2000) for an HMM and Wapiti (Lavergne *et al.*, 2010) for CRFs. The French treebank (Abeillé *et al.*, 2003) is used for training and evaluation of POS tagging based on a corpus split proposed in Seddah *et al.* (2013). We obtain 96.49% and 97.76% for the HMM and CRFs, respectively. Therefore, we use CRFs POS tagging results to preprocess the corpus for building POS-aware embedding, and accordingly the classification data set.

2.2 MWE-aware embedding

While multiword expressions (MWE) have been considered as a pain in the neck for natural language processing (Sag *et al.*, 2002), it also shows their importance in NLP. For French MWEs, Daille (2003) studied in the context of terminology, and Daille *et al.* (2004) and Morin & Daille (2010) presented MWEs in English-French translation alignment from the comparable corpus. Green *et al.* (2011) used tree substitution grammars to improve MWE identification and constituent parsing results. Dubremetz & Nivre (2014) created MWE data using the French Europarl corpus and corpus together with the lexicon of local grammars (Gross, 1975) for MWE detection by classification. Words in MWEs can give more senses when they are co-occurred instead of being processed separately. Therefore, we detect MWEs and deal with them as a single unit. For example, *au* from *au contraire* and *au cours de* are different in MWE-aware embedding because there are *au-contraire* and *au-cours-de* are also listed in addition to a single word *au*. Their distributions are calculated independently without considering *au* and *contraire*, or *au cours* and *de*.

The French treebank contains MWE annotation, in which phrase labels share the same label names with POS labels, for example, [**P** [P *D'*][P *après*]]. For MWE-aware embedding, we convert the treebank sentences into the sequence labeled sentences using the BIO format such as `B-MWE` and `I-MWE` as described in Figure 1. B-I-O stands for beginning-inside-outside of MWEs. We train bi-directional long-short-term-memory recurrent neural networks (bi-LSTM RNN) (Graves & Schmidhuber, 2005) using NeuroNER (Dernoncourt *et al.*, 2017b), and obtain up to 79.75% F1 score (79.47% in average for 5 runs). The French treebank is still used for training and evaluation of MWE detection. We note that we also train and evaluate MWE data using CRFs with ± 2 word and POS context information as a feature set, in which we obtain only 74.61% F1 score. Therefore, we use bi-LSTM RNN MWE detection to preprocess the corpus, and put together words of MWEs as a single unit for building MWE-aware embedding.

Wikipedia	Europarl	New crawls	Common crawl	Giga	in-domain	total
675M	64M	223M	89M	793M	664M	2.5B

TABLE 1 – The size (# of token) of corpora for embedding

2.3 Lemma-aware embedding

French is a moderately inflected language. [1] For example, a verb *avoir* ('have') can be inflected depending on person, number, mood, and tense : *ai*, *eus*, *avais*, *aurai*, *aie*, *eusse*, *aurais* for the first-person singular. Previously, Flemm used a rule-based method for lemmatization with POS-tagged results (Namer, 2000). L*efff* is a semi-automatically developed morphological and syntactic lexicon for French (Sagot, 2010) and there are systems based on L*efff* for lemmatization. We use a canonical form of the word (lemma) as in its basic unit of word embeddings, and we refer it as lemma-aware embedding. In lemma-aware embedding, different inflected forms such as *ai*, *eus*, *avais* are equally dealt with as *avoir*.

For lemma-aware embedding, we use a pipeline of TreeTagger (Schmid, 1994) and Flemm (Namer, 2000) for lemmatization. For unknown lemmas, especially for proper nouns, we directly use the surface form. Since there is currently no available evaluation data for lemmatization, we do not evaluate lemmatization results. The original French treebank contains such lemmatization information in the XML format. For example, a word *a* is annotated as a verb along with morphosyntactic properties including its lemma : `<w cat="V" ee="V-P3s" ei="VP3s" lemma="avoir" mph="P3s" subcat="">a</w>`. We leave the evaluation of lemmatization for future work.

2.4 Building word embeddings

The corpora for embedding include as follows :

1. Wikipedia, Europarl, and News Crawls for monolingual French,
2. French-side from the parallel corpus such as Common Crawl and Giga French-English, and
3. our in-domain corpus (described in § 3.1).

The size of these corpora (number of tokens) is summarized in Table 1. We use symbol normalization and tokenization schemes for French in Moses (Koehn *et al.*, 2007). Then, we post-edit tokenization errors such as a misuse of the apostrophe character. Since there are several tokenization error cases especially for a contraction, we build heuristic regular expressions to correct them. For sentence boundaries, we use TreeTagger's sentence boundary detection. All characters are lowercased for embeddings. Figure 2 presents sentence examples of the initial baseline, MWE-aware, and lemma-aware corpus for different embeddings. We build 300 dimension skip-gram embeddings with default options : 0.05 learning rate, and 5 for the size of the context window, etc.

Technically, we do not deal with proposed optimization methodologies putting together because they are contradictory. While POS-aware and MWE-aware embedding techniques increase sparsity by giving more number of vocabulary in the vector, lemma-aware embedding vector is intended to avoid sparsity.

1. https://en.wikipedia.org/wiki/French_language, accessed on January 12, 2018.

Initial :	*... l' ␣ arbitraire ␣ de ␣ la ␣ démesure ␣ veut ␣ susciter ␣ à ␣ tout ␣ prix ␣ .*
POS-aware :	*... l'/D ␣ arbitraire/N ␣ de/P ␣ la/D ␣ démesure/N ␣ veut/V ␣ susciter/V ␣ à/P ␣ tout/D ␣ prix/N ␣ ./PONCT*
MWE-aware :	*... l' ␣ arbitraire ␣ de ␣ la ␣ démesure ␣ veut ␣ susciter ␣ à-tout-prix ␣ .*
Lemma-aware :	*... le ␣ arbitraire ␣ de ␣ le ␣ démesure ␣ vouloir ␣ susciter ␣ à ␣ tout ␣ prix ␣ .*

FIGURE 2 – Example of the embedding corpus

3 Sentence Classification

We extrinsically evaluate the proposed optimization methods for word embedding by using the sentence classification task. First, we build sentence classification data, and then present classification results using various embedding settings and corpus sizes. We also explore the domain adaptation approach by using the in-domain corpus to build embedding vectors.

3.1 Sentence classification data and in-domain corpus

We build the sentence classification data set for French using opportunities : by opportunities, we deal with financial contract opportunities appeared in the municipal debriefing report. We download municipal debriefing reports from the city council all over France.

Our in-domain corpus consists of 886K documents and 664M tokens from these municipal debriefing reports. For classification, about 4,000 sentences are manually annotated either positive or negative to build the classification model. The classification data set is described in detail in Park *et al.* (2018). As described, all data sets are POS-annotated for POS-aware embedding, MWE-detected for MWE-aware embedding, and lemmatized for lemma-aware embedding experiments.

3.2 Classification experiments

For evaluation, we use several different embedding settings : baseline (based on inflected forms without POS labels, MWE detection or lemma), POS-aware (P-embedding), MWE-aware (M-embedding) and lemma-aware (L-embedding). We also build the embedding vector using the different size of the corpus : IN using the in-domain *relatively* small corpus and OUT using the large out-domain corpus from various sources for a domain adaptation approach. We use fastText (Joulin *et al.*, 2016) for building word embedding and classification. For embeddings, we build 300 dimension skip-gram models with default options. For classification, we use 1.0 for the learning rate, 25 epochs, and proposed pre-trained word vectors.

Table 2 shows classification results (accuracy) based on the different configurations alongside the size of vocabularies in embeddings. For a comparison purpose, we perform sentence classification using a pre-trained embedding vector for French provided by Bojanowski *et al.* (2017) (WIKI). It is a 300 dimension skip-gram model as ours. We also report a classification result "without" word embeddings (NONE) to show the effects of embedding vectors in classification.

Based on experiment results, we find following several practical facts. Using word embeddings improves classification results for all cases. Classification results are improved as the size of voca-

	IN	OUT	WIKI	NONE
baseline embedding	0.918 (0.4M)	0.909 (1.3M)	0.906 (1.1M)	0.901
P-embedding	0.919 (0.5M)	0.914 (1.8M)	-	0.912
M-embedding	0.920 (0.6M)	0.911 (1.7M)	-	0.909
L-embedding	0.901 (0.3M)	0.896 (1.1M)	-	0.891

TABLE 2 – Results based on the different embedding setting and the corpus size. We also provide the size of vocabularies. NONE is for the classification result without word embeddings. All data sets are POS-annotated, MWE-detected, and lemmatized for each experiment.

	IN	OUT	WIKI	NONE
baseline embedding	0.937	0.934	0.934	0.929
P-embedding	0.937	0.935	-	0.930
M-embedding	0.942	0.937	-	0.934
L-embedding	0.932	0.931	-	0.927

TABLE 3 – Results based on the bigram feature. All data sets are POS-annotated, MWE-detected, and lemmatized for each experiment as before.

bularies increase. Especially, "in-domain" information plays an important role regardless of its size in embeddings. Even with a small number of vocabularies and the small size of the training corpus, in-domain embeddings always outperform out-domain embeddings. [2]

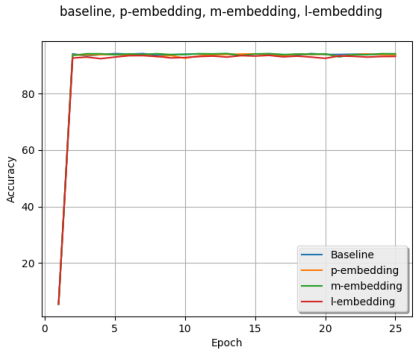
Results also show that context information matters to improve overall results such as in MWE-embeddings. This finding indicates that considering context as in MWE-aware embedding affect classification results. Therefore, co-occurred words should be processed together.

Finally, since context information matters, we extend experiments using a simple bigram feature. Wang & Manning (2012) already used bigram features to improve classification results for naive Bayes and support vector machines. Table 3 shows classification results with the bigram feature. Figure 3 shows results for each epoch using the bigram feature. While results for all embedding are converged in a very early stage, M-embedding can yield better results for almost all epochs (Figure 3a). We also compare classification results using embedding by in- and out-domain copora for m-embeddings (Figure 3b) where embeddings with the in-domain corpus outperform the out-domain corpus.

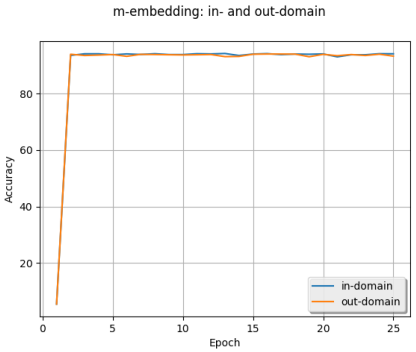
4 Discussion and Conclusion

We proposed several methods for optimizing word embeddings for French by using POS tagging, MWE detection and lemmatization. We used these embedding vectors in the sentence classification task, and MWE-aware in which the size of vocabularies are relatively large to other embeddings improved the classification result. To extend current MWE-aware embedding, we can consider formulaic sequences and named entities (Brooke *et al.*, 2017) in addition to MWEs. We leave these extensions for future work. While reported results show a minor improvement, they confirm our

2. We note that Fabre *et al.* (2014) also used the small size of the specialized corpus, which is similar to our in-domain corpus embeddings.



(a) Comparison of different embeddings



(b) Comparison of in-/out-domain corpora

FIGURE 3 – Results of each epoch : (a) for POS-aware (P-embedding), MWE-aware (M-embedding) and lemma-aware (L-embedding) using the in-domain corpus and the bigram feature, and (b) for MWE-aware (M-embedding) using the in/out-domain corpora and the bigram feature

intuition that incorporated context information in embeddings either linguistically motivated (MWEs) or not (bigram features) is important. As described, "in-domain" information played an important role, which should be well adapted to the proposed task. In-domain embeddings using a small number of vocabularies and the small size of the training corpus (roughly a third of the out-domain corpus) outperformed out-domain embeddings for all cases. The word embedding vectors, preprocessed source text files including the preprocessing script, and the sentence classification data for French are publicly available at `https://github.com/jungyeul/taln2018`.

Remerciements

We would like to thank the anonymous reviewers for their suggestions and comments.

Références

ABEILLÉ A., CLÉMENT L. & TOUSSENEL F. (2003). Building a Treebank for French. In *Treebanks*, p. 165–188. Kluwer.

AMMAR W., MULCAIRE G., TSVETKOV Y., LAMPLE G., DYER C. & SMITH N. A. (2016). Massively Multilingual Word Embeddings. *http ://arxiv.org/abs/1602.01925*.

ARORA S., LI Y., LIANG Y., MA T. & RISTESKI A. (2016). Linear Algebraic Structure of Word Senses, with Applications to Polysemy. *http ://arxiv.org/abs/1601.03764*.

BOJANOWSKI P., GRAVE E., JOULIN A. & MIKOLOV T. (2017). Enriching Word Vectors with Subword Information. *Transactions of the Association for Computational Linguistics*, **5**, 135–146.

BRANTS T. (2000). TnT – A Statistical Part-of-Speech Tagger. In *Proceedings of the Sixth Conference on Applied Natural Language Processing*, p. 224–231, Seattle, Washington, USA : Association for Computational Linguistics.

BRILL E. (1995). Transformation-Based-Error-Driven Learning and Natural Language Processing : A Case Study in Part-of-Speech Tagging. *Computational Linguistics*, **21**(4), 543–566.

BROOKE J., SNAJDER J. & BALDWIN T. (2017). Unsupervised Acquisition of Comprehensive Multiword Lexicons using Competition in an n-gram Lattice. *Transactions of the Association for Computational Linguistics*, **5**, 455–470.

CHEN X., XU L., LIU Z., SUN M. & LUAN H. (2015). Joint learning of character and word embeddings. In *Proceedings of the Twenty-Fourth International Joint Conference on Artificial Intelligence (IJCAI 2015)*, p. 1236–1242, Buenos Aires, Argentina : AAAI Press.

CHOI S., KIM T., SEOL J. & LEE S.-G. (2017). A Syllable-based Technique for Word Embeddings of Korean Words. In *Proceedings of the First Workshop on Subword and Character Level Models in NLP*, p. 36–40, Copenhagen, Denmark : Association for Computational Linguistics.

CHRUPAŁA G. (2013). Text segmentation with character-level text embeddings. In *Proceedings of ICML 2013 workshop on Deep Learning for Audio, Speech and Language Processing*, Atlanta, GA.

DAILLE B. (2003). Conceptual Structuring through Term Variations. In *Proceedings of the ACL 2003 Workshop on Multiword Expressions : Analysis, Acquisition and Treatment*, p. 9–16, Sapporo, Japan : Association for Computational Linguistics.

DAILLE B., DUFOUR-KOWALSKI S. & MORIN E. (2004). French-English Multi-word Term Alignment Based on Lexical Context Analysis . In *Proceedings of the Fourth International Conference on Language Resources and Evaluation (LREC'04)*, p. 919–922, Lisbon, Portugal : European Language Resources Association (ELRA).

DERNONCOURT F., LEE J. Y. & SZOLOVITS P. (2017a). Neural Networks for Joint Sentence Classification in Medical Paper Abstracts. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics : Volume 2, Short Papers*, p. 694–700, Valencia, Spain : Association for Computational Linguistics.

DERNONCOURT F., LEE J. Y. & SZOLOVITS P. (2017b). NeuroNER : an easy-to-use program for named-entity recognition based on neural networks. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing : System Demonstrations*, p. 97–102, Copenhagen, Denmark : Association for Computational Linguistics.

DUBREMETZ M. & NIVRE J. (2014). Extraction of Nominal Multiword Expressions in French. In *Proceedings of the 10th Workshop on Multiword Expressions (MWE)*, p. 72–76, Gothenburg, Sweden : Association for Computational Linguistics.

ERHAN D., BENGIO Y., COURVILLE A., MANZAGOL P.-A., VINCENT P. & BENGIO S. (2010). Why Does Unsupervised Pre-training Help Deep Learning ? *Journal of Machine Learning Research*, **11**, 625–660.

FABRE C., HATHOUT N., SAJOUS F. & TANGUY L. (2014). Ajuster l'analyse distributionnelle à un corpus spécialisé de petite taille. In *Actes de la 2ème édition de l'atelier SemDis (SemDis 2014)*, p. 266–279, Marseille, France.

FERRÉ A. (2017). Représentation de termes complexes dans un espace vectoriel relié à une ontologie pour une tâche de catégorisation. In *Actes de Rencontres des Jeunes Chercheurs en Intelligence Artificielle (RJCIA 2017)*, Caen, France.

GONZÁLEZ-GALLARDO C.-E. & TORRES-MORENO J.-M. (2017). Sentence Boundary Detection for French with Subword-Level Information Vectors and Convolutional Neural Networks. In *Proceedings of the International Conference on Natural Language, Signal and Speech Processing (ICNLSSP 2017)*, p. 80–84, Casablanca, Morocco.

GRAVES A. & SCHMIDHUBER J. (2005). Framewise phoneme classification with bidirectional LSTM and other neural network architectures. *Neural Networks*, **18**(5-6), 602–610.

GREEN S., DE MARNEFFE M.-C., BAUER J. & MANNING C. D. (2011). Multiword Expression Identification with Tree Substitution Grammars : A Parsing tour de force with French. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*, p. 725–735, Edinburgh, Scotland, UK. : Association for Computational Linguistics.

GROSS M. (1975). *Méthodes en syntaxe*. Hermann.

HE Z., CHEN W., LI Z., ZHANG M., ZHANG W. & ZHANG M. (2018). SEE : Syntax-aware Entity Embedding for Neural Relation Extraction. In *Proceedings of The Thirty-Second AAAI Conference on Artificial Intelligence (AAAI-18)*, New Orleans, Louisiana.

JOULIN A., GRAVE E., BOJANOWSKI P., DOUZE M., JÉGOU H. & MIKOLOV T. (2016). FastText.zip : Compressing text classification models. *http ://arxiv.org/abs/1612.03651*.

KOEHN P., HOANG H., BIRCH A., CALLISON-BURCH C., FEDERICO M., BERTOLDI N., COWAN B., SHEN W., MORAN C., ZENS R., DYER C., BOJAR O., CONSTANTIN A. & HERBST E. (2007). Moses : Open Source Toolkit for Statistical Machine Translation. In *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics Companion Volume Proceedings of the Demo and Poster Sessions*, p. 177–180, Prague, Czech Republic : Association for Computational Linguistics.

KUPIEC J. (1992). Robust part-of-speech tagging using a hidden Markov model. *Computer Speech and Language*, **6**(3), 225–242.

LAFFERTY J. D., MCCALLUM A. & PEREIRA F. C. N. (2001). Conditional Random Fields : Probabilistic Models for Segmenting and Labeling Sequence Data. In *Proceedings of the Eighteenth International Conference on Machine Learning*, ICML '01, p. 282–289, San Francisco, CA, USA : Morgan Kaufmann Publishers Inc.

LAVERGNE T., CAPPÉ O. & YVON F. (2010). Practical Very Large Scale CRFs. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, p. 504–513, Uppsala, Sweden : Association for Computational Linguistics.

LEVY O. & GOLDBERG Y. (2014). Dependency-Based Word Embeddings. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 2 : Short Papers)*, p. 302–308, Baltimore, Maryland : Association for Computational Linguistics.

LUONG T., SOCHER R. & MANNING C. D. (2013). Better Word Representations with Recursive Neural Networks for Morphology. In *Proceedings of the Seventeenth Conference on Computational Natural Language Learning*, p. 104–113, Sofia, Bulgaria : Association for Computational Linguistics.

MIKOLOV T., SUTSKEVER I., ANOOP D., HAI-SON L., STEFAN K. & JAN C. (2012). Subword Language Modeling with Neural Networks.

MIKOLOV T., SUTSKEVER I., CHEN K., CORRADO G. S. & DEAN J. (2013). Distributed Representations of Words and Phrases and their Compositionality. In C. J. C. BURGES, L. BOTTOU, M. WELLING, Z. GHAHRAMANI & K. Q. WEINBERGER, Eds., *Advances in Neural Information Processing Systems 26*, p. 3111–3119. Curran Associates, Inc.

MORIN E. & DAILLE B. (2010). Compositionality and lexical alignment of multi-word terms. *Language Resources and Evaluation*, **44**(1-2), 79–95.

NAMER F. (2000). Un analyseur Flexionnel de Français à base de règles. *Revue TAL*, **41**(2), 523–548.

PANG B. & LEE L. (2008). Opinion mining and sentiment analysis. *Foundations and Trends in Information Retrieval*, **2**(1-2), 1–135.

PARK J., DELAHAYE E. & BOVYN R. (2018). Detectio : Web Interface for Building Sentence Classification and User Recommendation Data. In *Proceedings of the 15th ESWC 2018 (INDUSTRY TRACK)*, Crete, Greece.

PENNINGTON J., SOCHER R. & MANNING C. D. (2014). GloVe : Global Vectors for Word Representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, p. 1532–1543, Doha, Qatar : Association for Computational Linguistics.

PINTER Y., GUTHRIE R. & EISENSTEIN J. (2017). Mimicking Word Embeddings using Subword RNNs. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, p. 102–112, Copenhagen, Denmark : Association for Computational Linguistics.

RATNAPARKHI A. (1996). A Maximum Entropy Model for Part-Of-Speech Tagging. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, p. 133–142, Philadelphia, Pennsylvania, USA.

SAG I. A., BALDWIN T., BOND F., COPESTAKE A. A. & FLICKINGER D. (2002). Multiword Expressions : A Pain in the Neck for NLP. In *Proceedings of the Third International Conference on Computational Linguistics and Intelligent Text Processing*, CICLing '02, p. 1–15, London, UK, UK : Springer-Verlag.

SAGOT B. (2010). The Lefff, a Freely Available and Large-coverage Morphological and Syntactic Lexicon for French. In *Proceedings of the Seventh conference on International Language Resources and Evaluation (LREC'10)*, Valletta, Malta : European Language Resources Association (ELRA).

SCHMID H. (1994). Probabilistic Part-of-Speech Tagging Using Decision Trees. In *Proceedings of International Conference on New Methods in Language Processing*, Manchester, UK.

SEDDAH D., TSARFATY R., KÜBLER S., CANDITO M., CHOI J. D., FARKAS R., FOSTER J., GOENAGA I., GOJENOLA GALLETEBEITIA K., GOLDBERG Y., GREEN S., HABASH N., KUHLMANN M., MAIER W., NIVRE J., PRZEPIÓRKOWSKI A., ROTH R., SEEKER W., VERSLEY Y., VINCZE V., WOLIŃSKI M., WRÓBLEWSKA A. & DE LA CLERGERIE E. V. (2013). Overview of the SPMRL 2013 Shared Task : A Cross-Framework Evaluation of Parsing Morphologically Rich Languages. In *Proceedings of the Fourth Workshop on Statistical Parsing of Morphologically-Rich Languages*, p. 146–182, Seattle, Washington, USA : Association for Computational Linguistics.

SOCHER R., HUVAL B., MANNING C. D. & NG A. Y. (2012). Semantic Compositionality through Recursive Matrix-Vector Spaces. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, p. 1201–1211, Jeju Island, Korea : Association for Computational Linguistics.

STRATOS K. (2017). Reconstruction of Word Embeddings from Sub-Word Parameters. In *Proceedings of the First Workshop on Subword and Character Level Models in NLP*, p. 130–135, Copenhagen, Denmark : Association for Computational Linguistics.

WANG S. & MANNING C. D. (2012). Baselines and Bigrams : Simple, Good Sentiment and Topic Classification. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (Volume 2 : Short Papers)*, p. 90–94, Jeju Island, Korea : Association for Computational Linguistics.

WIETING J., BANSAL M., GIMPEL K. & LIVESCU K. (2016). Charagram : Embedding Words and Sentences via Character n-grams. In *Proceedings of the 2016 Conference on Empirical Methods*

in Natural Language Processing, p. 1504–1515, Austin, Texas : Association for Computational Linguistics.

YU S., KULKARNI N., LEE H. & KIM J. (2017). Syllable-level Neural Language Model for Agglutinative Language. In *Proceedings of the First Workshop on Subword and Character Level Models in NLP*, p. 92–96, Copenhagen, Denmark : Association for Computational Linguistics.

Word2Vec vs LSA pour la détection des erreurs orthographiques produisant un dérèglement sémantique en arabe

Chiraz Ben Othmane Zribi[1]
(1) Laboratoire RIADI-GDL, ENSI, Université La Manouba, 2010, La Manouba
Chiraz.zribi@ensi-uma.tn

RESUME

Les mots en arabe sont très proches lexicalement les uns des autres. La probabilité de tomber sur un mot correct en commettant une erreur typographique est plus importante que pour le français ou pour l'anglais. Nous nous intéressons dans cet article à détecter les erreurs orthographiques plus précisément, celles générant des mots lexicalement corrects mais causant un dérèglement sémantique au niveau de la phrase. Nous décrivons et comparons deux méthodes se basant sur la représentation vectorielle du sens des mots. La première méthode utilise l'analyse sémantique latente (LSA). La seconde s'appuie sur le modèle Word2Vec et plus particulièrement l'architecture Skip-Gram. Les expérimentations ont montré que Skip-Gram surpasse LSA.

ABSTRACT

Word2Vec vs LSA for detecting semantic errors in Arabic language.
Arabic words are lexically close to each other. The probability of having a correct word by making a typographical error is greater than for French or English. We are interested in this article to detect spelling errors more precisely, those generating lexically correct words but causing a semantic disturbance in the sentence. We describe and compare two word embedding based methods. The first one uses Latent Semantic Analysis (LSA). The second, is based on the Word2Vec model and more precisely the Skip-Gram architecture. Experiments have showed that LSA is more efficient than Skip-Gram for both precision and recall.

MOTS-CLES : Erreurs orthographiques, dérèglement sémantique, représentation vectorielle, LSA, Word2Vec, Skip-Gram, langue arabe.

KEYWORDS : Spelling Errors, semantic disturbance, word embedding, LSA, Word2Vec, Skip-Gram, Arabic language.

1 Introduction

Les erreurs orthographiques qui produisent des mots lexicalement corrects causant un dérèglement sémantique au sein du contexte où elles se trouvent peuvent être dues à des problèmes de performance (i.e. faute de frappe) ou à des problèmes d'ignorance (i.e. confusion avec un autre mot).

Quand l'erreur est due à une faute de frappe par exemple, le mot erroné est généralement proche lexicalement du mot correct, comme dans les exemples amusants suivants :

Exemple en français : erreur de substitution d'une lettre par une autre :

La maman prépare un bon ***râteau*** (gâteau)

Exemple en arabe : interversion de deux lettres adjacentes

ترك له والده **ثورة** (ثروة)
/trk lh wAldh **vwrp** (vrwp)/
Son père lui a laissé une **révolution** (fortune)

Ces erreurs dites « sensibles au contexte », comptent environ 40% parmi toutes les erreurs orthographiques étudiées, selon (Verberne, 2002). Cette valeur assez importante a rendu l'étude de ce genre d'erreurs une nécessité en soi. En effet, plusieurs recherches ont été entreprises dans le but de remédier à ce problème notamment pour les langues indo-européennes telles que le français et l'anglais. Toutefois, en arabe, rares sont les travaux qui se sont attelés à les traiter en dépit de l'importance de cette tâche. En effet, les mots arabes sont lexicalement proches les uns des autres. Le risque de tomber sur un mot correct en commettant une erreur typographique (ajout d'une lettre, suppression d'une lettre, substitution d'une lettre par une autre et interversion de deux lettres adjacentes) est relativement important comme l'ont montré (Ben Othmane Zribi et al., 2005). Selon ces auteurs le nombre moyen de formes voisines qui diffèrent d'une seule opération d'édition est de 26,5 pouvant atteindre un maximum de 185, valeur importante comparée à celle calculée pour la langue française égale à 3,5 et celle relative à l'anglais égale à 3. Aussi, les auteurs nous renseignent sur la probabilité d'obtenir un mot correct lorsqu'une erreur est commise sur un mot. Cette probabilité pour un mot arabe (5,79%) est 10 fois plus grande que pour un mot anglais (0,59%) et 14 fois plus grande que pour un mot français (0,39%).

Nous nous focalisons dans cet article sur la détection des erreurs orthographiques en arabe qui produisent des mots lexicalement et syntaxiquement corrects mais qui causent des incohérences sémantiques. Nous utilisons et comparons à cet effet deux méthodes vectorielles qui permettent d'inférer le sens des mots à partir de leur distribution les uns par rapport aux autres, à savoir LSA (Landauer, 1998) et Skip-Gram (Mikolov et al., 2013). LSA est une méthode très utilisée pour représenter le sens, elle a été déjà utilisée pour détecter ce type d'erreurs avec quelques variantes au niveau de l'implémentation. La méthode Word2Vec, apparue ces dernières années, permet quant à elle de créer des vecteurs de sens utilisant des réseaux de neurones. Ceci a pour principal avantage de faciliter l'utilisation de grandes quantités de données d'apprentissage. Beaucoup de travaux ont été conduits utilisant Word2Vec mais, à notre connaissance, cette méthode n'a pas été auparavant testée pour détecter le type d'erreurs que nous visons. Afin de restreindre les champs de nos investigations, nous émettons l'hypothèse de l'existence d'une seule erreur sémantique par phrase et par mot. Cette erreur consisterait en une seule faute typographique générant un mot lexicalement et syntaxiquement correct, relevant de l'une des opérations d'édition citées précédemment. Des statistiques ont en effet montré que l'une (seulement) de ces opérations est à l'origine d'une erreur orthographique dans 90% des cas (Ben Hamadou, 1993). Nous avons également considéré l'Arabe moderne standard non voyellé car les écrits arabes sont généralement dépourvus de voyelles, c'est le cas des textes fréquemment rencontrés dans les journaux, les revues, les romans, etc. L'arabe voyellé concerne seulement quelques ouvrages poétiques ou littéraires didactiques ou encore le coran.

Le plan de l'article est comme suit : La section 2 est consacrée à la présentation de l'état de l'art. Nous décrivons dans la section 3 les deux méthodes que nous proposons. Nous présentons par la suite les expérimentations et les résultats obtenus dans la section 4. Enfin, nous concluons et donnons quelques perspectives dans la dernière section.

2 Etat de l'art

Dans la littérature, le problème des erreurs sémantiques a été considéré selon deux visions. Certains chercheurs ont considéré ce problème comme une tâche de résolution d'ambiguïté lexicale. Ils utilisent des ensembles de mots préétablis nommés "ensembles de confusion", contenant des mots semblables par le son (i.e. {stationary, stationery}), par l'écriture (i.e. {dessert, desert}) et par l'usage (i.e. {between, among}). Selon cette approche, un mot est simplement soupçonné lorsqu'un membre de son ensemble de confusion est mieux adapté à son contexte. Ce mot est corrigé en sélectionnant l'alternative la plus probable par rapport au contexte. (Golding, 1995) est à l'origine des méthodes basées sur le jeu de confusion. Il a proposé avec ses collègues plusieurs méthodes d'apprentissage automatique (pour le même ensemble de confusion), présentées ici dans l'ordre chronologique : la méthode hybride bayésienne basée sur les probabilités ainsi que les collocations (Golding, 1995), la méthode Tribayes combinant une méthode trigramme avec une méthode hybride bayésienne (Golding & Schabes, 1996) et l'algorithme de Winnow utilisant les mots voisins et adjacents avec un vote à majorité pondérée (Golding , Roth, 1999). Ces méthodes ont donné respectivement un taux de précision de 83%, 89% et 93,5% en moyenne. Le meilleur résultat a été obtenu plus tard par (Carlson et al., 2001). Ils ont proposé une méthode basée sur l'architecture d'apprentissage SNOW (un classificateur multi-classes) et ont testé jusqu'à 265 ensembles de confusion avec une précision de 99%. D'autres chercheurs les ont rejoints, comme par exemple le cas de (Mangu & Brill, 1997). Ils ont proposé de nouvelles méthodes et testé leur système sur les mêmes ensembles de confusion. Plus récemment, certains travaux ont développé des systèmes de correction basés sur les modèles n-gram à l'échelle du Web. Dans ces systèmes, le choix du mot dépend de la fréquence à laquelle chaque candidat (un membre de l'ensemble de confusion) a été vu dans le contexte donné dans des données d'apprentissage du Web, comme le Google N-gram Corpus. Nous pouvons citer par exemple (Bergsma et al., 2010) qui ont amélioré la précision (95,7% en moyenne) pour 5 ensembles de confusion dont la performance moyenne rapportée dans (Golding, Roth, 1999) est inférieure à 90%.

D'autres chercheurs ne se sont pas restreints à des ensembles de confusion prédéfinis. Ils ont utilisé le contexte pour détecter les erreurs sémantiques en appliquant des méthodes basées sur des informations sémantiques ou probabilistes. Les résultats obtenus sont souvent moins bons, car la tâche est plus difficile. Nous pouvons citer (Verberne, 2002) qui a appliqué une méthode trigramme et l'a testée sur 5 500 mots du British National Corpus (un sous-ensemble des données d'apprentissage) avec 606 erreurs. Les taux de rappel et de précision de détection sont respectivement de l'ordre de 72% et de 98%. Lorsque cette méthode a été testée sur des données de test hors entraînement les résultats pour la détection ont été largement inférieurs avec un taux de rappel de 51% et un taux de précision de 5%. (Hirst, Budanitsky, 2005) ont utilisé des mesures de distance sémantique dans WordNet pour détecter et corriger les malapropismes. Une erreur est signalée lorsqu'une variante d'orthographe (tout mot dont la distance d'édition est 1 du mot d'origine) entraîne un nouveau mot sémantiquement plus proche au contexte. Cette méthode a atteint une précision d'environ 23% lorsqu'elle a été testée sur environ 300 000 mots du corpus Wall Street Journal de 1987-89, avec environ 1 400 malapropismes introduits aléatoirement à une fréquence d'environ un mot sur 200. Plus récemment, (Zesch, 2012) a combiné une méthode statistique utilisant le modèle n-gram avec une méthode à base de connaissances utilisant WordNet, inspirée de celle de (Hirst, Budanitsky, 2005) pour détecter les malapropismes en anglais et en allemand. La combinaison des deux méthodes s'est révélée avantageuse au niveau des taux de précision qui sont de l'ordre de 90%. Les taux de rappel sont par contre faibles, ils sont d'environ 50% en moyenne. Aussi, pour détecter les erreurs sémantiques, (Gutierrez et al., 2014) a proposé une méthode basée sur le raisonnement logique utilisant une ontologie du domaine. Les taux de F-mesure obtenus varient entre 58% et 90%.

Pour l'arabe, un seul travail, à notre connaissance, s'est intéressé à la détection des erreurs sémantiques. (Ben Othmane Zribi, Ben Ahmed, 2013) ont proposé un Système Multi-Agents (SMA) combinant quatre méthodes contextuelles dont LSA et n'utilisant pas d'ensemble de confusion. Un système de vote permet de décider de la présence d'une erreur au sein d'une phrase. Les taux de précision et de rappel rapportés pour environ 1400 erreurs sémantiques générées artificiellement sont respectivement de l'ordre de 90% et de 83%.

3 Une méthode à base de représentation vectorielle des mots pour détecter les erreurs sémantiques

La majorité des chercheurs, en s'intéressant au problème des erreurs orthographiques sensibles au contexte, ont utilisé des ensembles de confusion. Les résultats obtenus sont d'une manière générale très satisfaisants car la problématique est relativement simple. Nous avons choisi dans ce travail de ne pas utiliser d'ensembles de confusion et de détecter toute erreur générant une incohérence sémantique au sein de son contexte. Ce choix est doublement motivé. D'une part, nous avons voulu traiter le problème des erreurs sémantiques dans sa globalité et ne pas nous limiter à un ensemble restreint d'erreurs prédéfinies. D'autre part, nous pensons qu'utiliser des ensembles de confusion pour l'arabe ne serait pas très judicieux. En effet, due à la proximité lexicale des mots, les ensembles de confusion seraient nombreux et de taille importante.
Afin de détecter ces erreurs sémantiques, nous faisons appel à deux méthodes se basant sur la représentation vectorielle des mots, à savoir LSA et Skip-Gram. Le modèle vectoriel n'est pas récent, il a en effet été introduit par (Salton et al., 1975) en recherche documentaire. Sa réhabilitation dans les recherches en TALN est par contre relativement récente notamment avec l'apparition des plongements lexicaux (word embeddings). Cette technique correspond à la représentation des mots par des vecteurs de nombres réels qui capturent leurs sens, leurs liens sémantiques et les différents contextes de leur utilisation. La représentation vectorielle des mots est principalement utilisée pour comparer les mots entre eux. Elle a ceci de particulier que les mots apparaissant dans des contextes similaires, et donc liés sémantiquement, possèdent des vecteurs correspondants qui sont relativement peu distants dans l'espace vectoriel où ils sont définis.
Chacune des deux méthodes proposées fournit sa propre représentation vectorielle des mots en fonction de leurs contextes. Nous calculons pour chaque mot à vérifier un coefficient de « validité sémantique » obtenu en comparant le vecteur de ce mot à tous les autres vecteurs-mot de la phrase. Un mot est soupçonné d'erreur si son coefficient de validité sémantique est inférieur à un seuil (déterminé empiriquement et préalablement établi), autrement dit, s'il est jugé suffisamment distant de ses voisins.

Afin déterminer le coefficient de validité sémantique d'un mot $\Omega(m_i)$ nous calculons la moyenne des distances angulaires entre le vecteur mot Vm_i et tous les $n-1$ vecteurs-mot dans la phrase (n étant le nombre de mots de la phrase) tout en privilégiant les mots contextuels les plus proches par rapport aux mots contextuels les plus éloignés.
Soit et Cg = {m_1, …, m_{i-1}} et Cd = {m_{i+1}, …, m_n} respectivement le contexte gauche et droit du mot à analyser m_i :

$$\overline{D_i} = \frac{\sum_{j=1}^{i-1} \frac{1}{i-j} D(Vm_i, Vmg_j) + \sum_{j=i+1}^{n} \frac{1}{j-i} D(Vm_i, Vmd_j)}{n-1}$$

$$\text{avec} \quad D(Vm_i, Vm_j) = \arccos \frac{Vm_i \bullet Vm_j}{\|Vm_i\| \times \|Vm_j\|}.$$

Cette distance est interprétée comme suit : “Deux mots *x* and *y* sont sémantiquement proches si *D(Vx, Vy)* ≤ 45 °. Lorsque *D (Vx, Vy)* > 45 °, la proximité sémantique est faible et pour 90° *x* et *y* n’ont aucune relation” (Schutze, 1998).

3.1 Principe et application de la méthode LSA

L’analyse sémantique latente (*LSA*) est l’une des méthodes les plus utilisées pour représenter le sens des mots. Elle permet d’identifier la similarité sémantique entre deux mots, deux segments textuels ou la combinaison des deux même si ces mots ou segments textuels ne sont pas co-occurrents. LSA prend en entrée un corpus textuel d’entrainement, construit une matrice de co-occurrences dont les lignes correspondent aux unités lexicales et les colonnes aux unités textuelles. Une normalisation est d’abord appliquée afin de réduire les poids des mots qui sont fréquents mais non informationnels. Ensuite, une réduction des dimensions de l’espace vectoriel des mots est réalisée à l’aide d’une analyse factorielle appelée décomposition en valeurs singulières.

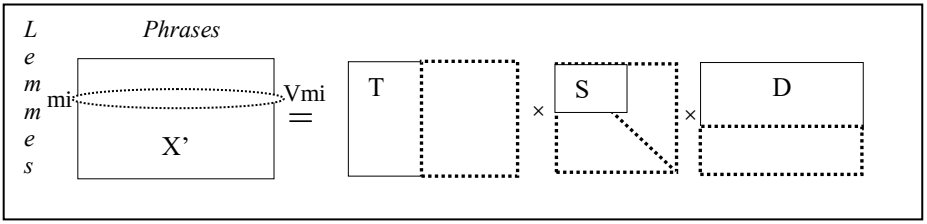


FIGURE 1 : Matrice de co-occurrence réduite

Dans ce travail, les vecteurs mots de la phrase à vérifier sont obtenus à partir de la matrice de cooccurrence réduite dont les lignes correspondent aux lemmes et les colonnes aux phrases. La taille de l’espace vectoriel est fixée à 300, valeur déterminée empiriquement. Le choix de la phrase comme contexte nous a paru raisonnable vu que celle-ci représente une unité sémantique dont le contenu se dégage du rapport établi entre les mots qu’elle contient.

3.2 Principe et application de l’architecture skip-gram du modèle Word2Vec

(Mikolov et al., 2013) considèrent que les méthodes déterministes basées sur le calcul des fréquences, telles que LSA, sont limitées pour représenter le sens des mots et ont introduit le modèle « Word2vec » à la communauté du TAL. Ce modèle est prédictif car il affecte des probabilités aux mots et a montré son efficacité par rapport à l’état de l’art pour des tâches de calcul de similarité et d’analogie entre les mots. Il est capable d’effectuer des tâches, comme le fameux exemple, vec(Roi)-vec(Homme)+vec(Femme) = vec(Reine), qui est un résultat assez intéressant. Il se base sur l’utilisation d’un réseau de neurones entrainé par des exemples de mots et de leurs contextes à partir d’un corpus d’apprentissage. Une fois entrainé, la transformation linéaire apprise au niveau de la couche cachée constitue la représentation vectorielle du mot cible. Le modèle Word2vec a été proposé en deux versions CBOW et Skip-Gram. CBOW permet de prédire un mot à partir d’un contexte tandis que Skip-gram prédit un contexte pour un mot. Pour notre problème, nous pensons que l’architecture Skip-Gram est plus adéquate, vu que nous vérifions pour chaque mot dans la phrase sa validité sémantique au sein de son contexte.

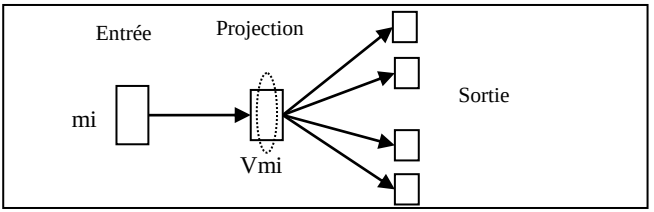


FIGURE 2 : Architecture Skip-Gram

Pareillement que pour LSA, nous avons fixé la taille des vecteurs mots à 300, ce qui correspond aux nombres de nœuds de la couche cachée. En outre, puisque LSA telle que nous l'avions définie tient compte du contexte de la phrase, nous avons choisi de faire en sorte que la taille de la fenêtre soit dynamique et toujours égale à la taille de la phrase en cours de traitement.

4 Expérimentations et comparaison des résultats

A cause de la non disponibilité de corpus contenant des erreurs naturelles correspondants à des mots appartenant au lexique, les travaux s'intéressant à la détection de ces erreurs ont été évalués dans leur majorité sur de erreurs générées de manière artificielle et introduites dans des corpus. Ne disposant pas d'un tel corpus pour l'arabe et dans le souci de comparer notre travail avec ce qui a été déjà proposé pour traiter ce type d'erreurs dans cette langue, nous avons utilisé les mêmes corpus d'apprentissage et de test que (Ben Othmane Zribi, Ben Ahmed, 2013). Rappelons que ce travail consiste en l'application d'un SMA combinant quatre méthodes contextuelles dont LSA. Signalons toutefois que la méthode LSA que nous utilisons considère la même taille de l'espace vectoriel (k = 300) mais diffère de cette dernière par le choix des unités lexicales car elle utilise le contexte de la « macro-phrase »[1] alors que nous considérons le contexte de la phrase simple. Le corpus d'apprentissage est composé d'un ensemble d'articles dans le domaine de l'économie[2] extraits du journal égyptien Al-Ahram (2009-2010) contenant environ 1 million de mots. Le corpus de test est extrait du même journal, mais hors apprentissage. Il contient environ 300 000 mots et 1 398 erreurs orthographiques produisant des dérèglements sémantiques générées artificiellement et introduites à la fréquence d'une erreur chaque 200 mots. Ces erreurs ont été générées semi-automatiquement et diffèrent des mots à remplacer d'une seule opération d'édition (insertion, suppression ou substitution d'un seul caractère, ou encore transposition de deux caractères adjacents). Nous avons utilisé le détecteur-correcteur d'erreurs orthographiques de (Ben Othmane & Zribi, 1999) qui fournit, en lui soumettant un mot correct, tous les mots lexicalement proches d'une seule opération d'édition. Les erreurs sont choisies manuellement parmi ces mots proches en faisant en sorte qu'elles ne soient pas des mots outils, qu'elles soient correctes syntaxiquement et qu'elles engendrent une incohérence sémantique au sein de la phrase. Voici un exemple d'erreur sémantique insérée dans notre corpus de test :

...ويتطلب المشروع قرضا من **الحنك** (البنك) الدولي ...
/…wytTlb Alm$rwE qrDA mn **AlHnk** (Albnk) Aldwly…/
… et le projet nécessitera un crédit **du palais** (bancaire) international …

[1] Macro-phrase est une phrase obtenue à l'issue d'un découpage à base de ponctuations et non de délimiteurs lexicaux. Elle peut correspondre à tout un paragraphe en français.
[2] Ce type de corpus est écrit en arabe standard moderne, plus facile à traiter que l'arabe classique qui est relativement plus ancien.

Comme l'illustre le tableau ci-dessous, notre méthode LSA avec un contexte restreint au niveau des unités lexicales, donne de meilleurs résultats au niveau de la précision que LSA avec un contexte plus élargi avec certes une petite perte au niveau du rappel. Elle reste cependant supérieure au niveau de la F-mesure. La méthode SMA est supérieure à notre méthode LSA aussi bien au niveau de la précision que de la F-mesure et ceci s'explique par la combinaison des quatre méthodes utilisées qui créent selon l'auteur une certaine synergie. En ce qui concerne Skip-Gram, nous remarquons d'abord sa supériorité à LSA aussi bien pour la précision que pour le rappel. Ensuite, comparativement à la méthode SMA, Skip-Gram est légèrement inférieure au niveau de la précision. Ceci pourrait s'expliquer par le fait que la méthode SMA utilise un processus de vote entre ses méthodes pour décider d'une erreur, ce qui la rend moins sensible au bruit. Cependant, Skip-Gram donne de meilleurs résultats globalement au niveau de la F-mesure avec une hausse de 5 points au niveau du rappel. Néanmoins, cette supériorité reste à vérifier en faisant par exemple varier la taille du corpus d'apprentissage. Nous pouvons citer (Altszyler, 2016) qui ont montré que LSA pouvait dépasser Skip-Gram quand le corpus est de petite taille.

Méthode	Précision (%)	Rappel (%)	F-mesure (%)
LSA (Ben Othmane Zribi et al., 2013)	79,62	84,44	81,96
SMA (Ben Othmane Zribi et al., 2013)	**90,55**	82,73	86,46
LSA	85,33	83,78	84,54
Skip-Gram	89,48	**86,15**	**87,78**

TABLE 1 : Evaluation et comparaison de la détection des erreurs

4.1 Un exemple d'échec

L'exemple suivant illustre un exemple d'erreur détectée à tort (faux positif) par les deux méthodes LSA et Skip-Gram :

لضمان لحاق الاقتصاد **بركب** الانتعاش وبناء القوة الاقتصادية الحقيقية

/… lDmAn lHAq AlAqtSAd **brkb** AlAntEA$ wbnA' Alqwp AlAqtSAdyp AlHqyqyp…/

… pour assurer que l'économie rattrape le **convoi** de la relance et la construction de la vraie force économique …

L'expression "لحاق...بركب"/lHAq …brkb /(rattraper le convoi) est une collocation non contiguë. Les mots de la phrase n'ont pas de lien sémantique avec le terme "ركب "/ rkb /(convoi), c'est pourquoi sa validité sémantique a été jugée faible.

5 Conclusion

Nous avons présenté dans cet article deux méthodes vectorielles afin de détecter les erreurs orthographiques causant un dérèglement sémantique au niveau de la phrase, à savoir LSA et Skip-Gram. La méthode Skip-Gram a donné des résultats encourageants par rapport à l'état de l'art. Elle a également montré sa supériorité par rapport à LSA aussi bien au niveau de la précision qu'au niveau du rappel. Nous comptons vérifier cette supériorité, dans le futur proche en faisant varier les tailles des corpus d'apprentissage et aussi le type des textes et voir dans quelle mesure le domaine des textes peut-il influer sur les résultats. Aussi, la correction automatique de ces erreurs en utilisant le contexte représente une perspective proche pour ce travail.

Références

ALTSZYLER E. (2016). Comparative study of LSA vs Word2vec embeddings in small corpora: a case study in dreams database. CoRR Journal abs/1610.01520.

BEN HAMADOU A. (1993). Vérification et correction automatique par analyse affixale des textes écrits, le cas de l'arabe non voyellé. Thèse d'état, Faculté des sciences de Tunis, 1993.

BEN OTHMANE ZRIBI C., BEN FRAJ F., BEN AHMED M. (2005). Un Système Multi-agent pour la Détection et la Correction des Erreurs Cachées en Langue Arabe. Actes de la 5ème conférence sur le Traitement Automatique des Langues Naturelles, Dourdan, France, 143-153.

BEN OTHMANE ZRIBI C., BEN AHMED M. (2013). Detection of semantic errors in Arabic texts. Artificial intelligence journal (195), 249-264.

BERGSMA S., PITLER E., LIN D. (2010). Creating Robust Supervised Classifiers via Web-Scale N-gram Data. Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics, Sweden, 865-874.

CARLSON A.J., ROSEN J., ROTH D. (2001). Scaling up Context-sensitive Text Correction. Proceedings of 13th Conference on Innovative Applications of Artificial Intelligence IAAI'01, Washington, USA, 45-50.

GOLDING A.R. (1995). A Bayesian hybrid method for context-sensitive spelling correction. Proceedings of the 3rd Workshop on Very Large Corpora, Massachusetts, USA, 39–53.

GOLDING A. R., SCHABES Y. (1996). Combining trigram based and feature based methods for context sensitive spelling correction, in: Proceedings of the 34th Annual Meeting of the Association for Computational Linguistics, Santa Cruz, 71-78.

GOLDING A.R., ROTH D. (1999). A winnow-based approach to context-sensitive spelling correction, Machine Learning journal, 34(1-3), 107-130.

GUTIERREZ F., DOU D., FICKAS S., GRIFFITHS G. (2014). Online Reasoning for Ontology-Based Error Detection in Text. OTM international conference on ontologies, databases and application of semantics, 562-579.

HIRST G., BUDANITSKY A. (2005). Correcting real-word spelling errors by restoring lexical cohesion. Natural Language Engineering (11), 87-111.

LANDAUER T. K., FOLTZ P.W. , LAHAM D.(1998). An introduction to Latent Semantic Analysis, Discourse Processes (25), 259-284.

MANGU L., BRILL E. (1997). Automatic Rule Acquisition for Spelling Correction, in: Proceedings of the 14th International Conference on Machine Learning, Nashville, 734-747.

MIKOLOV T., SUTSKEVER I., CHEN K., CORRADO G., DEAN J. (2103). Distributed representations of phrase and their compositionality. Advances in neural information processing systems, 3111-3119.

SALTON G., WONG A., YANG C. S. (1975). A vector space model for automatic indexing. Communications of the ACM (18), New York, NY, USA, 613-620.

SCHUTZE H. (1998). Automatic Word sense discrimination, Journal of Computational Linguistics (24), 97-123.

VERBERNE S. (2002). Context sensitive spell checking based on word trigram probabilities. Master thesis Taal, Spraak & Informatica, Nijmegen University.

ZESCH T. (2012). Detecting Malapropisms Using Measures of Contextual Fitness. Special Issue of the TAL Journal on “Managing Noise in the Signal: Error Handling in Natural Language Processing (53), 11-31.

Analyse de sentiments à base d'aspects par combinaison de réseaux profonds : application à des avis en français

Nihel Kooli Erwan Pigneul
PagesJaunes - Solocal, 125 boulevard Albert 1er, 35200 Rennes, France
nkooli@pagesjaunes.fr, epigneul@pagesjaunes.fr

RÉSUMÉ
Cet article propose une approche d'analyse de sentiments à base d'aspects dans un texte d'opinion. Cette approche se base sur deux étapes principales : l'extraction d'aspects et la classification du sentiment relatif à chaque aspect. Pour l'extraction d'aspects, nous proposons une nouvelle approche qui combine un CNN pour l'apprentissage de représentation de caractères, un b-LSTM pour joindre l'apprentissage de représentation de caractères et de mots et un CRF pour l'étiquetage des séquences de mots en entités. Pour la classification de sentiments, nous utilisons un réseau à mémoire d'attention pour associer un sentiment (positif, négatif ou neutre) à une expression d'aspect donnée. Les expérimentations sur des corpus d'avis (publics et industriels) en langue française ont montré des performances qui dépassent les méthodes existantes.

ABSTRACT
A combination of deep learning methods for aspect-based sentiment analysis : application to French reviews

This paper proposes an aspect based sentiment analysis approach in an opinioned text. The proposed method is composed of two main steps : aspect extraction and sentiment classification. The aspect extraction step uses a novel method that combines three processes : a CNN for the character level embedding, a b-LSTM to join the character and the word level embedding and a CRF for entity sequence tagging in the sentence. The sentiment classification process is based on a deep memory network to assign a sentiment polarity (positive, negative or neutral) to a given aspect target. The experiments on three French review datasets outperforms the state of the art.

MOTS-CLÉS : Analyse de sentiments à base d'aspects, Apprentissage de représentation, Étiquetage de séquences, Réseau de neurones convolutifs, Réseau récurrent à mémoire court et long terme bidirectionnel, Champs aléatoires conditionnels, Réseau à mémoire d'attention.

KEYWORDS: Aspect-based sentiment analysis, Word embedding, Entity sequence tagging, Convolutional Neural Network (CNN), bidirectional Long Short-Term Memory (b-LSTM), Conditional Random Fields (CRF), Deep memory network.

1 Introduction

L'analyse de sentiments à base d'aspects représente une tâche importante dans le domaine d'analyse de sentiments (Pang & Lee, 2008; Liu, 2012; Pontiki *et al.*, 2016). Il s'agit d'attribuer une polarité (positive, négative ou neutre) à chaque aspect évoqué dans une phrase d'opinion. Ceci est souvent réalisé par deux tâches principales : extraction d'aspects et analyse de sentiments niveau aspect.

L'extraction d'aspects consiste à identifier les aspects dans une phrase d'opinion donnée. Par exemple, dans l'avis "personnel sympathique, plats savoureux mais la note est chère", il s'agit d'extraire les expressions d'aspects : "personnel", "plats" et "note" et de leur associer respectivement les catégories : "service", "qualité" et "prix" .

L'analyse de sentiments niveau aspect consiste à associer un sentiment étant donné une expression d'aspect. Dans l'exemple ci-dessus, il s'agit d'attribuer un sentiment de polarité positive à "plats' et un sentiment de polarité négative à "note".

Nous présentons ci-dessous un aperçu sur les travaux existants pour résoudre ces deux tâches.

1.1 Extraction d'aspects

Dans la littérature, les méthodes d'extraction d'aspects sont souvent des méthodes à base de patrons linguistiques ou d'apprentissage supervisé.

Les approches à base de patrons linguistiques (Hu & Liu, 2004; Poria *et al.*, 2015) se basent principalement sur des règles grammaticales (et éventuellement sémantiques) et des dictionnaires lexicaux pour étiqueter et catégoriser les expressions d'aspect dans le texte d'opinion. Le problème de ce type d'approches est qu'elles nécessitent une pré-définition manuelle de ces règles par des experts, ce qui est très coûteux. De plus, ces règles sont dépendantes du langage et du domaine.

Les approches à base d'apprentissage supervisé traitent souvent le problème comme un problème d'étiquetage de séquence de mots en aspects. Les méthodes à base de champs aléatoires conditionnels (en anglais CRF), telle que celle proposée par Toh & Wang (2014), ont montré de bonnes performances. Le problème avec de telles méthodes est que le CRF est un modèle linéaire qui nécessite la définition manuelle d'un nombre important de caractéristiques.

Les méthodes à base d'apprentissage profond, qui sont des modèles appropriés pour traiter des données brutes, surmontent ces limites. En effet, ces méthodes ont montré récemment de très bonnes performances pour plusieurs tâches dans le domaine de TAL (Collobert *et al.*, 2011). À titre d'exemple, Poria *et al.* (2016) proposent une approche d'étiquetage des expressions d'aspects basée sur les réseaux de neurones convolutifs (en anglais CNN). Cette méthode dépasse les résultats obtenus par un CRF et prouve l'intérêt de l'emploi de l'apprentissage profond pour l'extraction d'aspects.

Dans cet article, nous proposons une nouvelle méthode d'extraction d'aspects qui combine plusieurs modèles : un CNN pour la représentation niveau caractère, un réseau récurrent à mémoire court et long terme bidirectionnel (en anglais b-LSTM) pour joindre la représentation niveau caractère et niveau mot et un CRF pour l'étiquetage du texte en aspects. Récemment, Ma & Hovy (2016) ont montré l'intérêt d'une telle combinaison pour d'autres tâches de TAL, notamment l'étiquetage morpho-syntaxique et la reconnaissance d'entités nommées.

1.2 Analyse de sentiments au niveau d'aspect

Le problème d'analyse de sentiments par rapport à une expression d'aspect donnée a été généralement traité comme un problème de classification supervisée en utilisant des modèles à base de caractéristiques prédéfinies ou des réseaux profonds. Le SVM s'est montré le modèle à base de caractéristiques prédéfinies le plus performant pour cette tâche. Nous citons, à titre d'exemple, Jiang *et al.* (2011) qui

proposent d'utiliser des caractéristiques syntaxiques, contextuelles et lexicales, telles que les mots clés, la ponctuation, l'étiquetage morpho-syntaxique, etc. Le problème de ces approches est qu'elles sont très dépendantes des caractéristiques manuellement définies.

Dans le cadre d'apprentissage profond, les LSTM (Tang *et al.*, 2015) ont montré de bonnes performances pour la classification de sentiments niveau aspect. À titre d'exempe, Tang *et al.* (2016a) proposent deux architectures de LSTM qui prennent en considération le mot d'aspect. La première (en anglais target dependent LSTM : TD-LSTM) modélise le contexte qui le précède et le suit de façon à ce qu'il soit pris en compte dans l'apprentissage de caractéristiques. Quant à la deuxième (en anglais target connection LSTM : TC-LSTM), elle prend en compte explicitement la connexion entre le mot d'aspect et chaque mot du contexte dans la phrase d'opinion. Les expérimentations ont prouvé l'intérêt de la prise en compte explicite de la connexion dans la représentation de la phrase d'opinion.

Wang *et al.* (2016) proposent un LSTM à base d'attention, qui offre un mécanisme d'attention permettant de se concentrer sur différentes parties de la phrase d'opinion étant donnés plusieurs aspects. La plongement (en anglais embedding) de l'expression d'aspects est prise en compte avec plongement de la séquence de mots pour attribuer des poids d'attention par rapport à un aspect donné à chaque mot.

Suite au succès du mécanisme d'attention dans ce contexte, Tang *et al.* (2016b) proposent un réseau à mémoire d'attention profond pour l'analyse de sentiment niveau aspect. Ce réseau est composé de plusieurs couches d'attention, qui emploient des informations sur le contenu et la localisation entre les mots, et qui partagent des paramètres de calcul. Cette approche dépasse les résultats de l'état de l'art sur l'analyse de sentiment dans des avis en Anglais.

Dans ce travail, nous proposons d'utiliser un réseau à mémoire d'attention combiné avec un apprentissage de plongement de mots basé sur les n-grammes pour prendre en compte les mots rares (Bojanowski *et al.*, 2016) dans le corpus.

Le reste de cet article est organisé comme suit. D'abord, la section 2 détaille notre nouvelle approche d'analyse de sentiments à base d'aspects. Ensuite, la section 3 reporte les expérimentations réalisées sur trois corpus d'avis en langue française. Enfin, la section 4 présente les conclusions et les perspectives de ce travail.

2 Méthode proposée

Nous proposons une nouvelle méthode d'analyse de sentiments à base d'aspects qui combine plusieurs réseaux profonds : CNN, b-LSTM et réseau à mémoire d'attention, et un modèle CRF. Cette méthode prend en entrée un texte d'opinion et propose d'extraire et catégoriser les expressions d'aspects. Un sentiment de polarité positive, négative ou neutre est ensuite attribué à chaque expression extraite. Dans le cas où plusieurs expressions référant à une même catégorie d'aspect occurrent dans le texte, la moyenne est calculée sur leurs sentiments associés.

Dans le reste de cette section, nous détaillons les deux principales étapes de notre approche.

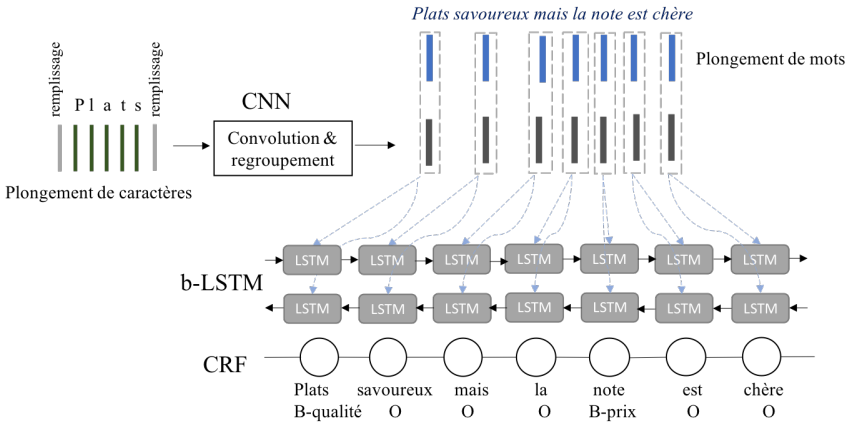


FIGURE 1 – Combinaison de CNN, B-LSTM et CRF pour l'extraction d'aspects

2.1 Extraction d'aspects

Le modèle global de l'approche d'extraction d'aspects est montré dans la FIGURE 1. D'abord une étape de représentation de mots à base de caractères est réalisée à l'aide d'un réseau CNN. Ensuite, cette représentation est combinée avec le plongement de mots à l'aide d'un b-LSTM. Enfin, le texte d'opinion est étiqueté en utilisant un CRF.

2.1.1 Représentation niveau caractère

Afin d'extraire les informations morphologiques (telles que le préfixe et le suffixe de mots) et de mieux représenter les mots d'une même famille, nous proposons d'apprendre une représentation de mots basée sur les caractères en utilisant un CNN, comme proposé par (Chiu & Nichols, 2015). En effet, il s'agit de générer un nouveau vecteur représentatif d'un mot en entrée en employant une couche de convolution suivie par une couche de max-regroupement (voir FIGURE 1). Pour ce faire, nous utilisons 30 filtres de taille 3 et 50 unités cachées.

2.1.2 Combinaison de représentation niveau mot et niveau caractère

Le plongement de mots est appris à l'aide d'une méthode basée sur les n-grammes, disponible sous la librairie Fasttext (Bojanowski *et al.*, 2016), sur les données de Wikipédia combinées avec un corpus privé d'avis en français. L'avantage de cette méthode est qu'elle permet de rajouter des informations sur les sous-mots offrant ainsi une représentation robuste des nouveaux mots qui n'apparaissent pas dans le corpus d'apprentissage. Nous utilisons ici des vecteurs de taille 100.

Cette représentation de mots est combinée avec la représentation niveau caractères à l'aide d'un b-LSTM. Ce dernier prend en entrée la concaténation des vecteurs de plongement de mots (obtenu à l'aide de Fasttext) et du vecteur de plongement de mots à base de caractères. Il permet de représenter

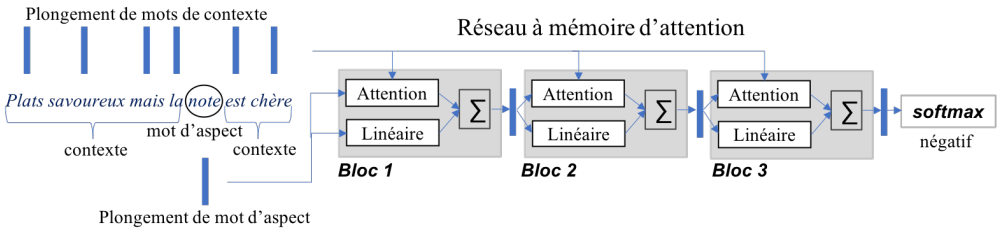


FIGURE 2 – Réseau à mémoire d'attention pour l'analyse de sentiments au niveau d'aspects

séparément la séquence de mots en avant et en arrière pour capturer les informations passées et futures, puis de les concaténer pour former la sortie finale (voir FIGURE 1).

Nous utilisons ici un b-LSTM de taille 200 et nous nous servons de l'algorithme RMSprop (Tieleman & Hinton, 2012) pour l'optimisation de la descente de gradient sous 50 itérations.

2.1.3 Étiquetage de séquence de mots

Nous proposons d'étiqueter le texte d'opinion en entrée par les entités correspondantes aux catégories d'aspects en utilisant le mode IOB2 (Sang & Veenstra, 1999). Pour ce faire, nous utilisons un modèle CRF qui permet de prendre en compte les relations entre les entités voisines dans l'étiquetage de la séquence de mots dans le texte d'opinion.

2.2 Analyse de sentiments au niveau d'aspects

Pour l'analyse d'aspects dans un texte d'opinion étant donné une expression d'aspect, nous proposons d'utiliser un réseau à mémoire d'attention (voir FIGURE 2), comme proposé par Tang *et al.* (2016b).

Ce réseau est composé d'une séquence de blocs, où chaque bloc est composé d'une couche d'attention et d'une couche linéaire. La couche d'attention permet de donner plus d'importance à certains mots de contexte que d'autres par rapport à un mot d'aspect en se basant sur la relation sémantique et de position entre mots. Par exemple, le mot "chère" est plus important que le mot "savoureux" pour le mot d'aspect "note" dans l'avis "Plats savoureux mais la note est chère". La couche linéaire est une transformation linéaire du vecteur de plongement de mot d'aspect. L'intérêt de cette couche a été démontré expérimentalement par Tang *et al.* (2016b).

L'entrée du réseau est la concaténation du plongement des mots de contexte (les mots à gauche et droite de l'expression d'aspect) dans le texte et celle du mot d'aspect (la moyenne dans le cas d'une expression d'aspects ayant plusieurs mots). Nous employons ici la même représentation utilisée dans l'étape d'extraction d'aspects. La sortie représente la polarité de sentiment associée à l'expression d'aspect fournie par la fonction *softmax*.

La succession de blocs permet d'augmenter le niveau d'abstraction. Le nombre de blocs est fixé empiriquement à 5. La descente de gradient est optimisée à l'aide de l'algorithme du gradient stochastique sous 150 itérations

3 Expérimentations

Nous avons testé notre approche sur trois corpus d'avis sur des restaurants et des musées en français (voir TABLE 1). Les deux premiers (Restaurants2 et Musées) sont des corpus publics de SemEval2016 (Apidianaki *et al.*, 2016), en éliminant les aspects implicites (dont l'expression d'aspect est représentée par *NULL*), et le troisième (Restaurants3) est un corpus industriel extrait du site "Pagesjaunes.fr". L'apprentissage est effectué sur un même corpus public de restaurants (Restaurants1) pour toutes les expérimentations. Six types d'aspects sont considérés pour ce corpus : service, prix, qualité, ambiance, localisation et général.

Corpus		# avis	# phrases	# aspects
Apprentissage	Restaurants1	337	1669	1797
Test	Restaurants2	120	696	708
	Musées	162	655	582
	Restaurants3	100	268	406

TABLE 1 – Corpus

Les résultats d'extraction d'aspects sont reportés dans la TABLE 2. Ce dernier montre que les performances obtenues en précision (69.73%), rappel (71.76%) et f1-mesure (70.73%) dépassent celles obtenues par l'approche basée sur les CNN (Poria *et al.*, 2016).

Ces résultats montrent également l'intérêt de l'apprentissage de la représentation niveau caractère et de l'emploi du modèle CRF pour l'étiquetage de séquences. Les cas d'échecs sont essentiellement dus à des erreurs orthographiques ou à un champ lexical très variable pour certains types d'expressions d'aspects (notamment, les aspects en relation avec la qualité de la cuisine).

Nous avons comparé notre approche de classification de sentiments à base d'aspects avec des méthodes de la littérature basées sur les LSTM, initialement proposés pour traiter des corpus en anglais et expérimenté ici sur nos trois corpus en français. Les résultats en taux de classification sont reportés dans la TABLE 3 et montrent la généricité des approches testées. En effet, malgré un apprentissage sur le corpus de restaurants, les résultats de tests sur le corpus de musées sont satisfaisants. De plus, ces résultats montrent que le réseau à mémoire d'attention maximise les taux de classification (74%) et confirment l'intérêt du mécanisme de mémoire d'attention.

Les cas d'échec sont essentiellement causés par des formules ironiques (par exemple,"*mais le summum fut l'arrivée des entrées*"), des formulations imagées (par exemple,"*une table qui tient la route dans*

Méthode	Restaurants2			Restaurants3		
	P (%)	R (%)	F1(%)	P (%)	R (%)	F1(%)
CNN (Poria *et al.*, 2016)	61.45	63.76	62.58	91.00	70.68	79.56
b-LSTM	64.68	69.62	67.06	91.07	71.50	80.10
CNN+b-LSTM	66.29	**72.06**	69.06	**93.71**	69.63	79.89
b-LSTM+CRF	**71.14**	70.38	70.76	92.12	71.03	80.21
CNN+b-LSTM+CRF	70.65	71.30	**70.97**	92.40	**73.83**	**82.08**

TABLE 2 – Résultats d'extraction d'aspects en Précision (P), Rappel (R) et F1-mesure (F1)

Méthode	Restaurants2	Musées	Restaurants3
LSTM (Tang *et al.*, 2015)	68.22	62.37	76.47
TD-LSTM (Tang *et al.*, 2016a)	66.99	60.48	81.62
TC-LSTM (Tang *et al.*, 2016a)	65.13	61.86	80.15
LSTM à base d'attention (Wang *et al.*, 2016)	69.74	64.95	80.88
Réseau à mémoire d'attention	**74.23**	**66.71**	**83.82**

TABLE 3 – Résultats d'analyse de sentiments relatifs aux aspects en taux de classification (%)

un quartier où les tables changent très vite ?") ou des formules complexes contenant des oppositions (par exemple,"*vraiment dommage vu le quartier et le cadre*") dans les commentaires.

4 Conclusion

Cet article propose une nouvelle approche d'analyse de sentiments à base d'aspects dans un texte d'opinion. Cette approche est une combinaison de plusieurs réseaux profonds : CNN, b-LSTM et réseau à mémoire d'attention, et un modèle CRF. Les expérimentations sur trois corpus d'avis en français ont montré des performances qui dépassent l'existant et ont prouvé l'intérêt de cette combinaison pour l'extraction d'aspects et pour l'analyse de sentiment relatif à chaque aspect extrait.

Comme perspectives de ces travaux, nous proposons de traiter les cas où les aspects ne sont pas explicitement mentionnés dans les avis. Par exemple, dans l'avis "Tout n'est pas fait maison mais c'est bon", l'aspect "qualité" est implicite dans la phrase. Un autre point à approfondir concerne la tolérance aux erreurs orthographiques. Enfin, nous envisageons d'expérimenter notre approche sur d'autres domaines, tels que des avis sur des médecins, des plombiers, des films, de la musique, etc.

Références

APIDIANAKI M., TANNIER X. & RICHART C. (2016). Datasets for aspect-based sentiment analysis in french. In N. C. C. CHAIR), K. CHOUKRI, T. DECLERCK, S. GOGGI, M. GROBELNIK, B. MAEGAARD, J. MARIANI, H. MAZO, A. MORENO, J. ODIJK & S. PIPERIDIS, Eds., *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, Paris, France : European Language Resources Association (ELRA).

BOJANOWSKI P., GRAVE E., JOULIN A. & MIKOLOV T. (2016). Enriching word vectors with subword information. *arXiv preprint arXiv :1607.04606*.

CHIU J. P. C. & NICHOLS E. (2015). Named entity recognition with bidirectional lstm-cnns. cite arxiv :1511.08308.

COLLOBERT R., WESTON J., BOTTOU L., KARLEN M., KAVUKCUOGLU K. & KUKSA P. (2011). Natural language processing (almost) from scratch. *J. Mach. Learn. Res.*, **12**, 2493–2537.

HU M. & LIU B. (2004). Mining and summarizing customer reviews. In *Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '04, p. 168–177, New York, NY, USA : ACM.

JIANG L., YU M., ZHOU M., LIU X. & ZHAO T. (2011). Target-dependent twitter sentiment classification. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics : Human Language Technologies - Volume 1*, HLT '11, p. 151–160, Stroudsburg, PA, USA : Association for Computational Linguistics.

LIU B. (2012). *Sentiment Analysis and Opinion Mining*. Morgan & Claypool Publishers.

MA X. & HOVY E. H. (2016). End-to-end sequence labeling via bi-directional lstm-cnns-crf. *CoRR*, **abs/1603.01354**.

PANG B. & LEE L. (2008). Opinion mining and sentiment analysis. *Found. Trends Inf. Retr.*, **2**(1-2), 1–135.

PONTIKI M., GALANIS D., PAPAGEORGIOU H., ANDROUTSOPOULOS I., MANANDHAR S., AL-SMADI M., AL-AYYOUB M., ZHAO Y., QIN B., DE CLERCQ O., HOSTE V., APIDIANAKI M., TANNIER X., LOUKACHEVITCH N., KOTELNIKOV E., BEL N., JIMÉNEZ-ZAFRA S. M. & ERYIĞIT G. (2016). Semeval-2016 task 5 : aspect based sentiment analysis. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*, p. 19–30 : Association for Computational Linguistics.

PORIA S., CAMBRIA E. & GELBUKH A. (2016). Aspect extraction for opinion mining with a deep convolutional neural network. *Know.-Based Syst.*, **108**(C), 42–49.

PORIA S., CAMBRIA E., GELBUKH A. F., BISIO F. & HUSSAIN A. (2015). Sentiment data flow analysis by means of dynamic linguistic patterns. *IEEE Comp. Int. Mag.*, **10**(4), 26–36.

SANG E. F. T. K. & VEENSTRA J. (1999). Representing text chunks. In *Proceedings of the Ninth Conference on European Chapter of the Association for Computational Linguistics*, EACL '99, p. 173–179, Stroudsburg, PA, USA : Association for Computational Linguistics.

TANG D., QIN B., FENG X. & LIU T. (2016a). Effective lstms for target-dependent sentiment classification. In *COLING*, p. 3298–3307 : ACL.

TANG D., QIN B. & LIU T. (2015). Document modeling with gated recurrent neural network for sentiment classification. In L. MÀRQUEZ, C. CALLISON-BURCH, J. SU, D. PIGHIN & Y. MARTON, Eds., *EMNLP*, p. 1422–1432 : The Association for Computational Linguistics.

TANG D., QIN B. & LIU T. (2016b). Aspect level sentiment classification with deep memory network. *CoRR*, **abs/1605.08900**.

TIELEMAN T. & HINTON G. (2012). Lecture 6.5—RmsProp : Divide the gradient by a running average of its recent magnitude. COURSERA : Neural Networks for Machine Learning.

TOH Z. & WANG W. (2014). DLIREC : aspect term extraction and term polarity classification system. In *SemEval@COLING*, p. 235–240 : The Association for Computer Linguistics.

WANG Y., HUANG M., ZHU X. & ZHAO L. (2016). Attention-based LSTM for aspect-level sentiment classification. In J. SU, X. CARRERAS & K. DUH, Eds., *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, EMNLP 2016, Austin, Texas, USA, November 1-4, 2016*, p. 606–615 : The Association for Computational Linguistics.

Predicting the Semantic Textual Similarity with Siamese CNN and LSTM

Elvys Linhares Pontes[1] Stéphane Huet[1] Andréa Carneiro Linhares[2]
Juan-Manuel Torres-Moreno[1,3]
(1) LIA, Université d'Avignon et des Pays de Vaucluse, Avignon, 84000 France
(2) Universidade Federal do Ceará, Sobral, Ceará Brazil
(3) École Polytechnique de Montréal, Montréal, Canada
{elvys.linhares-pontes, juan-manuel.torres,
stephane.huet}@univ-avignon.fr
andrea.linhares@ufc.br

RÉSUMÉ
La Similarité Textuelle Sémantique (STS) est la base de nombreuses applications dans le Traitement Automatique du Langage Naturel (TALN). Notre système combine des réseaux neuronaux convolutifs et récurrents pour mesurer la similarité sémantique des phrases. Il utilise un réseau convolutif pour tenir compte du contexte local des mots et un LSTM pour prendre en considération le contexte global d'une phrase. Cette combinaison des réseaux préserve mieux les informations significatives des phrases et améliore le calcul de la similarité entre les phrases. Notre modèle a obtenu de bons résultats et est compétitif avec les meilleurs systèmes de l'état de l'art.

ABSTRACT
Semantic Textual Similarity (STS) is the basis of many applications in Natural Language Processing (NLP). Our system combines convolution and recurrent neural networks to measure the semantic similarity of sentences. It uses a convolution network to take account of the local context of words and an LSTM to consider the global context of sentences. This combination of networks helps to preserve the relevant information of sentences and improves the calculation of the similarity between sentences. Our model has achieved good results and is competitive with the best state-of-the-art systems.

MOTS-CLÉS : Similarité, Réseaux de neurones siamois, LSTM, CNN.

KEYWORDS: Similarity, Siamese Neural Networks, LSTM, CNN.

1 Introduction

Semantic Text Similarity (STS) is an important task in Natural Language Processing (NLP) applications such as information retrieval, classification, extraction, question answering, and plagiarism detection. The STS task measures the degree of similarity between two texts and can be expressed as follows: given two sentences, a system returns a continuous score on a scale from 1 to 5, with 1 indicating that the semantics of the sentences are completely independent and 5 meaning that there is a semantic equivalence.

STS is a difficult issue since languages have numerous ambiguities and synonymous expressions, while sentences may have variable lengths and complex structures. Therefore basic models, e.g.

bag-of-words or TF-IDF models, are constrained by their specificities that put aside the role played by the word order and ignore syntactic as well as semantic relationships. Recent successes in sentence similarity have been obtained using Neural Networks (RNNs: Recurrent Neural Networks (Mueller & Thyagarajan, 2016; Kiros *et al.*, 2015; Tai *et al.*, 2015) and CNNs: Convolutional Neural Networks (He *et al.*, 2015)). Neural Networks (NNs) use a deep analysis of sentences and words to take better into account both the semantics and the structure of sentences in order to predict the sentence similarity.

In this paper, we describe our technique based on NNs to measure similarity. First, we use a Siamese CNN to analyze the local context of words in a sentence and to generate a representation of the relevance of a word and its neighborhood. Then, we use a Siamese LSTM to analyze the entire sentence based on its words and its local contexts. At last, we predict the semantic similarity of pairs of sentences using the Manhattan distance.

We applied our framework on the SemEval information for STS assignment and we acquired competitive outcomes demonstrating that our model can give helpful information to enhance the sentence analysis.

This paper is organized as follows: we make an overview of relevant work for STS in Section 2. Next, we detail our approach in Section 3. The experimental setup and results are presented in Sections 4 and 5, respectively. Finally, we give our conclusion and some last remarks in Section 6.

2 Related Work

To deal with the STS task, previous studies have resorted to various features (e.g. word overlap, synonym/antonym), linguistic resources (e.g. WordNet and pre-trained word embeddings) and a wide assortment of learning algorithms (e.g. Support Vector Regression (SVR), regression functions and NNs). Among these works, several techniques extract multiple features of sentences and apply regression functions to estimate these similarity scores (Lai & Hockenmaier, 2014; Zhao *et al.*, 2014; Bjerva *et al.*, 2014; Severyn *et al.*, 2013). Lai & Hockenmaier (2014) analyzed distinctive word relations (e.g. synonyms, antonyms, and hyperonyms) with features based on counts of co-occurences with other words and similarities between captions of images. Zhao *et al.* (2014) predicted the sentence similarity from syntactic relationship, distinctive content similitudes, length and string features. Bjerva *et al.* (2014) also utilized a regression algorithm to foresee the STS from different features (WordNet, word overlap, and so forth). Finally, Severyn *et al.* (2013) combined relational syntactic structures with SVR.

The development of NNs has improved the results of many NLP applications and especially the STS task (He *et al.*, 2015; Mueller & Thyagarajan, 2016; Tsubaki *et al.*, 2016; Rychalska *et al.*, 2016). Architectures such as RNNs and CNNs further improve the semantic analysis and the prediction of sentence relatedness.

RNNs differ from other NN models in their ability to process sequential information. They update a memory cell to make sense of data read in a sentence over time. Rychalska *et al.* (2016) used a Recursive AutoEncoder (RAE) and a WordNet grant framework to produce sentence embeddings. They consolidated these embeddings with a Support Vector Machine (SVM) classifier to compute a semantic relatedness score. Long Short Term Memory (LSTM) enhances RNNs to handle long-term dependencies (Mueller & Thyagarajan, 2016; Greff *et al.*, 2015; Tai *et al.*, 2015). The LSTM engineering is made out of a memory cell and non-direct gating units that update its state over time

and manage the data stream into/out the cell. Mueller & Thyagarajan (2016) used a Siamese LSTM to encode sentences using pre-trained word embedding vectors. Siamese LSTMs used the same weights to encode sentences and to produce comparable sentence representations for similar sentences. Then, they predicted the closeness of pair of sentences using the Manhattan distance between the sentence representations. Tai *et al.* (2015) introduced the Tree-LSTM that is a generalization of LSTM for tree-structured network topologies. They utilized this Tree-LSTM to encode a couple of sentences and to predict their closeness with a NN that analyzes the distance and the angle between the sentence embeddings.

CNNs have accomplished excellent outcomes in classification (Kim, 2014) and other NLP tasks (Collobert *et al.*, 2011). He *et al.* (2015) generated sentence embedding using a Siamese CNN architecture with various convolution and pooling operations to extract distinctive granularities of information. Their convolution uses filters that analyze entire word embeddings and each dimension of word embeddings with multiple window sizes. For output of the convolution operation, they applied several pooling types (max, mean, and min). Finally, they predicted the sentence similarity from numerous measurements (horizontal and vertical comparison) to compare local regions of sentence representation.

In this work, we join the ideas examined in (Mueller & Thyagarajan, 2016) and (Kim, 2014) to produce more accurate semantic sentence embeddings. The next section presents our model and its characteristics w.r.t. previous work.

3 Our model

A sentence is composed of words which can form phrases and clauses. Examining a sentence and its components helps us to comprehend its meaning. NNs are structures that can inspect relationships between words from multiple points of view. On the one hand, LSTMs can recognize and process the semantics of a sentence by investigating the words through time. They update their state to get the gist of the sentence (global context) in the order of words. In this procedure, LSTMs filter unimportant data by retaining just the main information. On the other hand, CNNs use layers with convolution filters that are connected to local features (Kim, 2014). They enable the analysis of a sentence from multiple perspectives (filters). This type of NNs does not have the same concern with the sentence length as LSTMs since CNNs examine all the words of the sentence together. Nonetheless, CNNs do not consider the order of words in their analysis, so these structures cannot investigate sequence relationships in the sentence.

Differently from (Mueller & Thyagarajan, 2016) that only analyze the general context of words and from (He *et al.*, 2015) that do not consider the order of words in the sentences, we analyze the words in two perspectives: general and local contexts. Words are considered through time from the general information of a word (word embedding) and its specific semantic and syntactic features (local context) based on its previous and its following words. We apply a CNN to investigate the local context for each word in a sentence. The CNN analyzes together all the words of the local context and generates their representation as a unique structure. Then, we utilize an LSTM to examine the words of the sentence one by one (Figure 1). Our NN has a Siamese structure (Mueller & Thyagarajan, 2016; He *et al.*, 2015), i.e. our CNN^A and our $LSTM^A$ are equal to our CNN^B and our $LSTM^B$, respectively. The following subsection describes our CNN, our LSTM, and our similarity metrics to predict the sentence similarity.

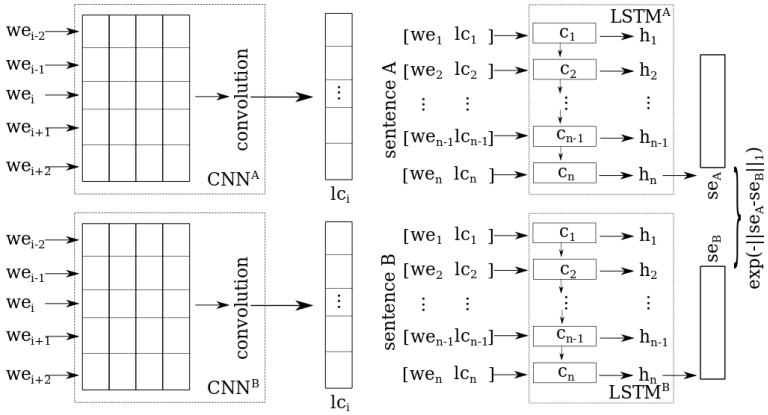


FIGURE 1 – Siamese CNN+LSTM to calculate the similarity of a pair of sentences.

3.1 Neural Network Architecture

Kim trained a simple CNN on top of pre-trained word vectors for the sentence classification task (Kim, 2014). His simple model composed of one layer of convolution achieved excellent results on multiple benchmarks. Inspired by the good results of CNNs in the sentence classification (Kim, 2014), we use a Siamese CNN to generate local contexts for each word in a sentence from its previous and following words. We utilize pre-trained word embeddings [1] to represent these words. Let $\mathbf{we}_i \in R^k$ be the k-dimensional word vector corresponding to the i-th word in a sentence. A local context of length l (e.g. $l = 5$) is represented as:

$$\mathbf{xl_i} = \mathbf{x_{i-2}} \oplus \mathbf{x_{i-1}} \oplus \mathbf{x_i} \oplus \mathbf{x_{i+1}} \oplus \mathbf{x_{i+2}} \tag{1}$$

where $\oplus$ is the concatenation operator. Our convolution operation involves a filter $w \in R^{lk}$, which is applied to a window of l words to produce a local context. In more details, our CNN generates the local context of word i by:

$$\mathbf{lc_i} = f(\mathbf{w} \cdot \mathbf{xl_i} + \mathbf{b}) \tag{2}$$

where $\mathbf{b}$ is a bias term and f is the hyperbolic tangent function. This filter is connected to every sequence of words in a sentence to deliver a local context for all words.

In order to analyze the general and the local contexts of the word i, we concatenate its pre-trained word embeddings $\mathbf{we_i}$ (general semantic and syntactic features that were learned on a large corpus) and its local context $\mathbf{lc_i}$. Our LSTM updates its state c_i and produces an output h_i at time step i in a sentence using the equations described in (Mueller & Thyagarajan, 2016). The last output of our LSTM h_n represents the meaning of a sentence.

Diverse similarity metrics (cosine, Euclidean and Manhattan distances) were tested and we acquired the best outcome with the Manhattan distance $\exp(-||se_A - se_B||) \in [0, 1]$. Since these scores are not optimized for the similarity metric range (1-5), we apply in a post-processing step a regression method using local regression and bandwidth to project our predictions in the correct scale, similarly to (Li & Racine, 2003).

1. Publicly available at: code.google.com/p/word2vec

4 Experimental Setup

We use the SICK dataset to analyze and to test the performance of our system. This dataset contains 9,927 sentence pairs (Marelli *et al.*, 2014) and we split it in 4,927/2,000/3,000 for training/validation/test. Each sentence pair is annotated with a relatedness label $\in [1, 5]$ corresponding to the average relatedness judged by 10 different individuals. The gold scores for relatedness are composed of: 923 pairs within the [1,2) range, 1,373 pairs within the [2,3) range, 3,872 pairs within the [3,4) range, and 3,672 pairs within the [4,5] range.

We initialize our CNN and our LSTM weights with small random Gaussian entries. Our CNN has filters R^{300} and our LSTM has 50-dimensional hidden representations $\boldsymbol{h}_t$ and memory cells $\boldsymbol{c}_t$. We use a forget bias of 2.5 to model long-range dependencies, Adadelta method to optimize the parameters, and a learning rate of 0.01. We did not identify any improvement with deep LSTMs because of the small amount of data. Like (Mueller & Thyagarajan, 2016), we also augmented our training dataset and we pre-trained our network using the dataset of SemEval 2013 STS task.

5 Results

In order to understand the relevance of the local context for the sentence similarity, we investigated the original Siamese LSTM without local context and compared it with our method using various lengths for the local context: 3, 5, 7, and 9 (Table 1). The original Siamese LSTM analyzes a sentence considering only the general context of words. As expected, the analysis of general and local contexts of words improved the sentence analysis, according to the Pearson's and Pearman's correlation coefficients and the Mean Squared Error (MSE) scores. Short or long local contexts did not generate the best results, which shows that short local context (3 words) did not get enough information about the neighborhood of words and long local context (7 words) includes irrelevant information.

Method	r	ρ	**MSE**
Siamese LSTM (Mueller & Thyagarajan, 2016)	*0.8822*	*0.8345*	*0.2286*
Siamese LSTM (publicly available version) [2]	0.8500	0.7860	0.3017
Siamese #local context: 3 + Siamese LSTM	0.8536	0.7909	0.2915
Siamese #local context: 5 + Siamese LSTM	**0.8549**	**0.7933**	**0.2898**
Siamese #local context: 7 + Siamese LSTM	0.8540	0.7922	0.2911
Siamese #local context: 9 + Siamese LSTM	0.8533	0.7890	0.2923
Non-Linear Similarity (Tsubaki *et al.*, 2016)	0.8480	0.7968	0.2904
Constituency Tree LSTM (Tai *et al.*, 2015)	0.8582	0.7966	0.2734
Skip-thought+COCO (Kiros et al. 2015)	0.8655	0.7995	0.2561
Dependency Tree LSTM (Tai *et al.*, 2015)	0.8676	**0.8083**	**0.2532**
ConvNets (He *et al.*, 2015)	**0.8686**	0.8047	0.2606

TABLE 1 – Pearson (r) and Spearman (ρ) correlation coefficients, and Mean Squared Error for the test set of STS task.

2. We used the public version of Siamese LSTM (Mueller & Thyagarajan, 2016) available at `https://github.com/aditya1503/Siamese-LSTM`, however, we did not get the same results as the ones described in their paper.

The bottom part of Table 1 compares the results of our system and the best state-of-the-art systems. Although our method did not generate the best results, our system is among the top systems and the results were improved with respect to the publicly available version of the original Siamese LSTM.

In order to illustrate how our local context acts on sentence analysis, Table 2 shows at the word level the similarity a pair of paraphrases: *"Her life spanned years of incredible change for women."* and *"Mary lived through an era of liberating reform for women."* For each pair of words taken in both sentences, the similarity measured as a cosine distance [3] is computed either from general word embeddings (table a) or local contexts of length 5 (table b). The first things to notice is that the two tables have different ranges of values because they each represent a different dimensional space; this means that values must be compared inside each table. Analyzing Table 2a shows that word embeddings preserve general semantic and syntactic relationships of words. In this case, the words are more similar to the words that have similar semantics (*1-"Her"*, *2-"Mary"* and *2-"women"*; *1-"life"* and *2-"lived"*; *1-"change"* and *2-"reform"*) and/or have similar syntactic roles (*1-"of"* and *2-"for"*). Table 2b highlights that the local context of a word has its semantic and syntactic features based on the words in its window; e.g. the nearest contexts to *1-"life"* are *2-"Mary"*, *2-"lived"*, *2-through* and *2-"women"* since these local contexts have directly (*2-"lived"*) and indirectly (*2-"Mary"*, *2-"through"* and *2-"women"*) similar semantics. This analysis is similar to the syntactic features for the local contexts, e.g. the nearest local context of *1-"for"* are *2-"lived"*, *2-"of"*, *2-"for"* and *2-"woman"*. The relevance of local context is strengthened when we analyze phrasal verbs or multi-word expressions in which meaning depends strongly on their previous and their following words.

	Mary	**lived**	**through**	**an**	**era**	**of**	**liberating**	**reform**	**for**	**women**
Her	0.77	0.93	0.90	0.81	1.04	0.92	0.95	0.91	0.80	0.80
life	0.91	0.70	0.89	0.90	0.82	1.00	0.71	0.86	0.88	0.86
spanned	0.88	0.76	0.81	1.01	0.80	0.85	0.92	1.00	0.89	0.93
years	0.88	0.70	0.94	0.88	0.72	0.86	0.92	0.93	0.81	0.86
of	0.93	0.96	0.96	1.09	0.91	0.00	0.99	1.02	0.82	0.91
incredible	0.94	0.89	0.83	0.94	0.84	0.95	0.74	1.04	0.83	0.97
change	0.97	0.90	0.93	0.92	0.85	0.99	0.80	0.67	0.83	0.92
for	0.96	0.97	0.67	0.79	0.89	0.82	0.88	0.92	0.00	0.89
women	0.81	0.96	0.99	0.93	0.92	0.91	0.79	0.88	0.89	0.00

a. Cosine distance between word embeddings.

	Mary	**lived**	**through**	**an**	**era**	**of**	**liberating**	**reform**	**for**	**women**
Her	0.06	0.08	0.09	0.11	0.16	0.12	0.13	0.13	0.09	0.08
life	0.10	0.08	0.09	0.12	0.11	0.13	0.13	0.14	0.10	0.10
spanned	0.15	0.14	0.11	0.11	0.18	0.14	0.14	0.16	0.13	0.12
years	0.13	0.11	0.08	0.13	0.10	0.12	0.11	0.16	0.09	0.09
of	0.12	0.11	0.10	0.12	0.11	0.09	0.12	0.14	0.13	0.11
incredible	0.12	0.12	0.13	0.14	0.19	0.13	0.03	0.16	0.14	0.09
change	0.14	0.13	0.18	0.15	0.18	0.15	0.16	0.02	0.15	0.13
for	0.10	0.09	0.10	0.11	0.12	0.08	0.11	0.12	0.04	0.08
women	0.09	0.07	0.09	0.11	0.11	0.08	0.09	0.14	0.07	0.01

b. Cosine distance between local contexts of length 5.

TABLE 2 – Cosine distance measured between word embeddings (a.) and between the local contexts of length 5 (b.) for each pair of words of two paraphrases.

3. The cosine distance between two vectors u and v is defined by $1 - \frac{u \cdot v}{||u||_2 ||v||_2}$.

Table 3 shows four examples of STS scores for multiple levels of similarities. The first pair of sentences describes an example of active and passive voice, with the same meaning (4.9 golden score). The second case is an example of positive and negative sentences (3.3 golden score). The third example is composed of sentences that do not share the same meaning, having 1.0 golden score. Finally, our method helps to determine the semantic relationship of the phrasal verb "*wipe off*" and the verb "*clean*" in the last example. Our approach improves the Siamese LSTM analysis by generating better scores. The local context helps to better identify not only similar sentences but also the negation and sentences with different meanings. This local information provides LSTM with a smoother analysis of words and how they connect in a sentence.

Pair of sentences	**Golden score**	**Siamese LSTM**	**Our approach**
Fish is being cooked by a woman. *A woman is cooking fish.*	4.9	3.84	**4.05**
The bearded man is not sitting on a train. *The bearded man is sitting on a train.*	3.3	3.49	**3.35**
Someone is playing with a toad. *The trumpet is being played by a man.*	1.0	1.51	**1.46**
I will wash up if you wipe off the table. *I will wash up if you clean the table.*	5.0	3.67	**4.08**

TABLE 3 – Examples of semantic textual similarities using Siamese LSTM and our approach (Siamese #local context: 5 + Siamese LSTM).

To sum up, the local context of words refined the general context analysis. Our approach identified more details about the words and their local as well as general contexts, which usually leads to improved STS scores.

6 Conclusion

STS is an important task for various NLP applications, e.g. Automatic Text Summarization (ATS), Question-Answering, Information Retrieval, etc. Our system combines CNN and LSTM structures to analyze, to identify and to preserve the relevant information in each part of sentences and in the whole sentences. The local context turned out to be useful to get complement information about a word in a sentence and to improve the sentence analysis. In our experiments, the local context improved the prediction of the sentence similarity, by reducing the mean squared error and increasing the correlation scores.

We plan to test other methods to analyze the local context (Ermakova & Mothe, 2016; Zhu *et al.*, 2017). Unfortunately, the dataset we used for the experiments is of a modest size and we did not find larger annotated corpora for this task. Therefore, we also want to lead extrinsic evaluations by measuring how STS acts on ATS systems, depending on whether the original or the modified Siamese LSTM model is used.

Acknowledgments

This work was partially financed by the European Project CHISTERA-AMIS ANR-15-CHR2-0001.

References

BJERVA J., BOS J., VAN DER GOOT R. & NISSIM M. (2014). The meaning factory: Formal semantics for recognizing textual entailment and determining semantic similarity. In *SemEval@COLING*, p. 642–646: The Association for Computer Linguistics.

COLLOBERT R., WESTON J., BOTTOU L., KARLEN M., KAVUKCUOGLU K. & KUKSA P. (2011). Natural language processing (almost) from scratch. *J. Mach. Learn. Res.*, **12**, 2493–2537.

ERMAKOVA L. & MOTHE J. (2016). Query expansion by local context analysis. In *CORIA 2016 - Conférence en Recherche d'Informations et Applications- 13th French Information Retrieval Conference. CIFED 2016 Colloque International Francophone sur l'Ecrit et le Document, Toulouse, France, March 9-11, 2016, Toulouse, France, March 9-11, 2016.*, p. 235–250.

GREFF K., SRIVASTAVA R. K., KOUTNÍK J., STEUNEBRINK B. R. & SCHMIDHUBER J. (2015). LSTM: A search space odyssey. *CoRR*, **abs/1503.04069**.

HE H., GIMPEL K. & LIN J. J. (2015). Multi-perspective sentence similarity modeling with convolutional neural networks. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, EMNLP 2015, Lisbon, Portugal, September 17-21, 2015*, p. 1576–1586.

KIM Y. (2014). Convolutional neural networks for sentence classification. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, EMNLP 2014, October 25-29, 2014, Doha, Qatar, A meeting of SIGDAT, a Special Interest Group of the ACL*, p. 1746–1751.

KIROS R., ZHU Y., SALAKHUTDINOV R., ZEMEL R. S., TORRALBA A., URTASUN R. & FIDLER S. (2015). Skip-thought vectors. In *Proceedings of the 28th International Conference on Neural Information Processing Systems*, NIPS'15, p. 3294–3302, Cambridge, MA, USA: MIT Press.

LAI A. & HOCKENMAIER J. (2014). Illinois-lh: A denotational and distributional approach to semantics. In *Proceedings of the 8th International Workshop on Semantic Evaluation, SemEval@COLING 2014, Dublin, Ireland, August 23-24, 2014.*, p. 329–334.

LI Q. & RACINE J. (2003). Nonparametric estimation of distributions with categorical and continuous data. *Journal of Multivariate Analysis*, **86**(2), 266–292.

MARELLI M., MENINI S., BARONI M., BENTIVOGLI L., BERNARDI R. & ZAMPARELLI R. (2014). A SICK cure for the evaluation of compositional distributional semantic models. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation, LREC 2014, Reykjavik, Iceland, May 26-31, 2014.*, p. 216–223.

MUELLER J. & THYAGARAJAN A. (2016). Siamese recurrent architectures for learning sentence similarity. In *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence*, AAAI'16, p. 2786–2792: AAAI Press.

RYCHALSKA B., PAKULSKA K., CHODOROWSKA K., WALCZAK W. & ANDRUSZKIEWICZ P. (2016). Samsung poland nlp team at semeval-2016 task 1: Necessity for diversity; combining recursive autoencoders, wordnet and ensemble methods to measure semantic similarity. In *SemEval@NAACL-HLT*.

SEVERYN A., NICOSIA M. & MOSCHITTI R. (2013). Learning semantic textual similarity with structural representations. In *In ACL*.

TAI K. S., SOCHER R. & MANNING C. D. (2015). Improved semantic representations from tree-structured long short-term memory networks. *CoRR*, **abs/1503.00075**.

TSUBAKI M., DUH K., SHIMBO M. & MATSUMOTO Y. (2016). Non-linear similarity learning for compositionality. In *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence, February 12-17, 2016, Phoenix, Arizona, USA.*, p. 2828–2834.

ZHAO J., ZHU T. & LAN M. (2014). ECNU: One stone two birds: Ensemble of heterogenous measures for semantic relatedness and textual entailment. In *Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014)*, p. 271–277: Association for Computational Linguistics.

ZHU Q., LI X., CONESA A. & PEREIRA C. (2017). Gram-cnn: a deep learning approach with local context for named entity recognition in biomedical text. *Bioinformatics*.

L'évaluation des représentations vectorielles de mots en utilisant WordNet

Nourredine Aliane[1] Jean-Jacques Mariage[1] Gilles Bernard[1,2]
(1) Laboratoire LIASD Université Paris 8, 2 rue de la Liberté 93526 Saint-Denis cedex
(2) Instutit d'Enseignement à Distance
nourredine@ai.univ-paris8.fr, jjm@ai.univ-paris8.fr,
gilles.bernard@iedparis8.net

RÉSUMÉ

Les méthodes d'évaluation actuelles des représentations vectorielles de mots utilisent généralement un jeu de données restreint et biaisé. Pour pallier à ce problème nous présentons une nouvelle approche, basée sur la similarité entre les synsets associés aux mots dans la volumineuse base de données lexicale WordNet. Notre méthode d'évaluation consiste dans un premier temps à classer automatiquement les représentions vectorielles de mots à l'aide d'un algorithme de clustering, puis à évaluer la cohérence sémantique et syntaxique des clusters produits. Cette évaluation est effectuée en calculant la similarité entre les mots de chaque cluster, pris deux à deux, en utilisant des mesures de similarité entre les mots dans WordNet proposées par NLTK (wup _similarity). Nous obtenons, pour chaque cluster, une valeur entre 0 et 1. Un cluster dont la valeur est 1 est un cluster dont tous les mots appartiennent au même synset. Nous calculons ensuite la moyenne des mesures de tous les clusters. Nous avons utilisé notre nouvelle approche pour étudier et comparer trois méthodes de représentations vectorielles : une méthode traditionnelle, WebSOM et deux méthodes récentes, word2vec (Skip-Gram et CBOW) et GloVe, sur trois corpus : en anglais, en français et en arabe.

ABSTRACT

Evaluating word representations using WordNet.

Current evaluation methods for word representations generally use a restricted and biased dataset. To overcome this problem we present a new approach, based on the similarity between synsets associated with words in the large WordNet lexical database. Our evaluation method consists first of all in automatically arranging the vector representations of words in clusters with a clustering algorithm, and then to evaluate the semantic and syntactic coherence of the clusters produced. This evaluation is performed by calculating the similarity between the words of each cluster, taken two by two, using similarity measures between the words in WordNet proposed by NLTK (wup _similarity). We obtain, for each cluster, a value Between 0 and 1. A cluster whose value is 1 is a cluster whose words belong to the same synset. The average of the measurements of all the clusters is then calculated. We used our new approach to study and compare three vector representation methods : a traditional WebSOM method and two recent methods, word2vec (Skip-Gram and CBOW) and GloVe, on three corpora : in English, French and Arabic.

MOTS-CLÉS : Representations vectorielles de mots, word2vec, GloVe, WebSOM, WordNet, similarité entre mots, clustering.

KEYWORDS: word vector representations, word2vec, GloVe, WebSOM, WordNet, word similarity, clustering.

1 Introduction

Une méthode de représentation vectorielle a pour but d'associer à chaque mot dans un corpus textuel, un vecteur à valeurs réelles, tel que les composantes de ce vecteur décrivent le mieux possible le sens de ce mot dans son contexte. Cependant, la tâche la plus difficile est de vérifier la qualité des vecteurs produits par ces méthodes. L'évaluation se fait généralement par un travail manuel ou avec une évaluation directe, en utilisant un petit jeu de données comme : "WordSim-353" (Finkelstein *et al.*, 2001), "TOEFL" (Landauer & Dutnais, 1997) constitué de 80 questions à choix multiples, "Google's analogy dataset" qui comporte 19 544 analogies ou "MSR's analogy dataset" qui contient 8000 analogies morpho-syntaxiques. Une étude récente de (Baroni *et al.*, 2014), conduit un ensemble d'expériences en comparant la méthode de word2vec (Mikolov *et al.*, 2013) aux autres méthodes traditionnelles. (Levy *et al.*, 2015) ont toutefois trouvé des résultats différents, en utilisant le même jeu de données. Comme le rappellent (Claveau & Kijak, 2015), "*L'évaluation directe séduit par sa simplicité, mais pose la question de l'adéquation des lexiques utilisés comme références*". Afin de permettre d'utiliser un lexique de référence plus conséquent, nous proposons une nouvelle approche d'évaluation indirecte, basée sur l'idée d'exploiter les mesures de similarités entre les mots de WordNet (Miller, 1995). Ces mesures sont présentées dans (Pedersen *et al.*, 2004). Elles sont implémentées dans NLTK [1]. La base de données lexicale WordNet (Miller, 1995), plus volumineuse, offre des possibilités plus intéressantes. Elle contient plus de 200 000 mots, avec leurs relations sémantiques et lexicales. Nous décrivons notre approche plus en détail dans la section 4. Pour nos expériences, nous avons choisi trois méthodes de représentations vectorielles de mots : word2vec (Mikolov *et al.*, 2013), GloVe (Pennington *et al.*, 2014) et WebSOM (Kohonen *et al.*, 1998). Elles sont toutes fondées sur l'hypothèse de (Harris, 1954), selon laquelle les mots apparaissant dans des contextes similaires ont un sens similaire. Les modèles de représentation vectorielle de mots transforment l'analyse distributionnelle d'un corpus en espace vectoriel, dans lequel deux vecteurs proches géométriquement représentent deux mots sémantiquement proches.

2 Représentation vectorielle de mots

Dans ce paragraphe, nous présentons brièvement les trois méthodes de representation vectorielle de mots que nous avons choisies pour conduire nos expérimentations.

2.1 WebSOM

Cette technique de représentation a été utilisée dans le système WebSOM (Kohonen *et al.*, 1998). Dans un premier temps, il est associé à chaque mot m_i, un vecteur x_i, de dimension d, dont les composantes sont des nombres réels, initialisés aléatoirement entre 0 et 1. Dans un second temps, le mot m_i sera représenté par un autre vecteur X_i, qui est déterminé de la façon suivante :

$$X_i = (p_N(x_i)...p_1(x_i)\ \epsilon.x_i\ s_1(x_i)...s_N(x_i))^T.$$

Où : $p_1(x_i)$ et $s_1(x_i)$ sont respectivement les vecteurs moyens des vecteurs qui correspondent à tous les mots prédécesseurs immédiats et successeurs immédiats du mot m_i dans l'ensemble de corpus, ($p_1(x_i)$ et $s_1(x_i)$ sont également de dimension d). La fenêtre contextuelle est de $(2N + 1)$ mots

1. NLTK (Natural Language Toolkit) est une bibliothèque logicielle en Python.

(le mot courant, N mots précédents et N mots suivants). ϵ est un réel positif inférieur à 1. Il sert à contrôler le rôle du mot m_i dans son contexte ($\epsilon = 1$ signifie que le vecteur x_i, initialisé aléatoirement au départ, et les autres vecteurs des contextes du mot m_i, sont de même importance). Le vecteur X_i est de dimension $(2N + 1)d$

2.2 word2vec

Une méthode proposée par (Mikolov *et al.*, 2013) fondée sur les réseaux de neurones, a été implémentée dans un outil qui s'appelle word2vec. Deux modèles de représentation des mots sont implémentés dans word2vec. Ce sont le continuous bag-of-words (CBOW) et le Skip-Gram.

1. CBOW a pour objectif de prédire la probabilité d'un mot, à partir de ses contextes. Cette représentation de mots consomme moins de temps en entraînement que le skip-gram.
2. Skip-Gram, contrairement aux CBOW, vise à prédire la probabilité des contextes d'un mot à partir de ce mot.

2.3 GloVe

GloVe (Global Vectors for Word Representation) (Pennington *et al.*, 2014) est un modèle proposé par l'équipe NLP de l'université de Stanford. Cette méthode combine les avantages de la factorisation matricielle globale et des méthodes de contexte local. Le contexte est une fenêtre de longueur fixe d'éléments lexicaux centrés sur le mot. On cherche à représenter chaque mot i et chaque mot j apparaissant dans le même contexte par des vecteur v_i et v_j respectivement, de dimension d tels que : $v_i.v_j + b_i + b_j = log(X_{ij})$. Où X_{ij} représente le nombre de fois où le mot j se produit dans le contexte du mot i. b_i et b_j sont des biais scalaires associés aux mots i et j respectivement.

3 WordNet et la similarité entre les mots

WordNet (Miller, 1995) est une grande base de données lexicale de l'anglais, développée par des linguistes de l'université de Princeton. Les mots y sont regroupés en ensembles de synonymes cognitifs (synsets). Les synsets sont interconnectés au moyen de relations conceptuelles-sémantiques et lexicales. WordNet est distribuée sous licence libre, la dernière version 3.1 répertorie plus de 200 000 mots. (Pedersen *et al.*, 2004) présentent plusieurs algorithmes de calcul de similarité entre mots, utilisant la structure et le contenu de WordNet. Après étude de ces différentes mesures de similarités, la wup_similarity semble la plus pertinante.

wup_similarity (Wu & Palmer, 1994) : renvoie un score entre 0 et 1, en fonction des profondeurs des deux mots et de celle de leur dernier ancêtre commun dans une taxonomie. Elle est définie par l'équation :

$$wup\_similarity(s_1, s_2) = \frac{2 * depth(lcs(s_1, s_2))}{depth(s_1) + depth(s_2)} \quad (1)$$

$lcs(s_1, s_2)$: le dernier ancêtre commun entre s_1 et s_2 (pour l'anglais : least common subsumer). Il correspond au dernier noeud du graphe taxonomique à partir duquel divergent les branches de s_1 et s_2. $depth(s_i)$ est la profondeur de s_i (le nombre d'arêtes entre la racine et s_i, $depth(racine) = 1$).

4 Méthodologie

La Figure 1 donne une vue générale de la méthode proposée :

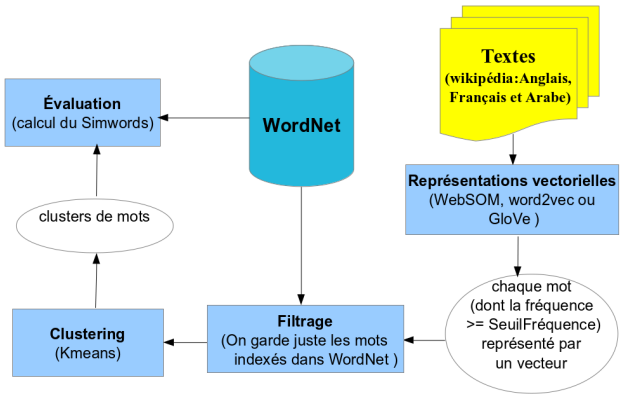


FIGURE 1 – Le principe de fonctionnement de notre système

Représentation vectorielle de mots : Nous appliquons les trois méthodes de représentation vectorielle de mots sur nos corpus, afin d'associer un vecteur à chaque mot dont la fréquence d'usage est supérieure à un seuil donné.

Filtrage : Cette étape consiste à sélectionner, parmi les mots qui ont été représentés par des vecteurs, ceux qui sont indexés dans WordNet ($w \in WordNet \equiv wordnet.synsets(w) \neq \emptyset$).

Clustering : Après la représentation vectorielle de mots, nous utilisons Kmeans++ (Arthur & Vassilvitskii, 2007) [2], afin de regrouper les mots qui ont une proximité sémantique ou syntaxique dans un même cluster.

Évaluation : Nous calculons la similarité entre les mots de chaque cluster, pris deux à deux en utilisant la $wup\_similarity$. Nous définissons $Simwords(C_i)$ [3] la similarité entre les mots du cluster C_i par l'équation (2).

2. kmeans++ est implémenté dans Scikit-learn (est une bibliothèque libre Python dédiée à l'apprentissage automatique).
3. https ://github.com/nourredinealiane/simwords.

$$Simwords(C_i) = \frac{\sum_{k=1}^{n_{C_i}-1} \sum_{j=k+1}^{n_{C_i}} wup\_simMax(m_k, m_j)}{n_{C_i}(n_{C_i}-1)/2} \quad (2)$$

$$wup\_simMax(m_k, m_j) = \underset{k \in wordnet.synsets(m_k),\ j \in wordnet.synsets(m_j)}{ArgMax}\ wup\_similarity(k, j) \quad (3)$$

Où les mots m_k et m_j appartiennent au clusters C_i. Nous calculons ensuite la moyenne des mesures de tous les clusters, par l'équation :

$$Simwords = \frac{\sum_{i=1}^{k} Simwords(C_i)}{k} \quad (4)$$

Où n_{C_i} est le nombre de mots du cluster C_i, k est le nombre de clusters.

Note : Pour l'arabe et le français, NLTK utilise une traduction automatique, par exemple :

wordnet.synsets(m, lang='fra'), renvoie les synonymes de la traduction en anglais du mot français m (pour l'arabe, *lang = 'arb'*). Le script Python ci-dessous montre comment calculer wup_simMax entre le mot «chat» et le mot «chien».

```
>>> from nltk.corpus import wordnet
>>> from itertools import product
>>> syns1 = wordnet.synsets('chat', lang='fra')
>>> syns1
[Synset('computerized_tomography.n.01'), Synset('cat.v.01'),
Synset('felis.n.01'), Synset('cat.n.01'), Synset('tom.n.02'),...]
>>> syns2 = wordnet.synsets('chien', lang='fra')
>>> syns2
[Synset('dog.n.01'), Synset('pooch.n.01'), Synset('hound.n.01'),
Synset('andiron.n.01'), Synset('pawl.n.01'),...]
>>> wup_simMax = max((wordnet.wup_similarity(s1, s2) or 0, s1, s2)
                              for s1, s2 in product(syns1, syns2))
>>> wup_simMax
(0.8571428571428571, Synset('cat.n.01'), Synset('dog.n.01'))
#le dernier ancetre commun est :
>>> wup_simMax[1].lowest_common_hypernyms(wup_simMax[2])
[Synset('carnivore.n.01')]
```

5 Corpus

Pour évaluer et comparer les trois méthodes de représentations vectorielles de mots choisies, nous avons sélectionné trois corpus textuels (Wikipédia dump 2017) en trois langues différentes : anglais, français et arabe. Les propriétés détaillées de ces corpus textuels sont indiquées dans le tableau 1.

Corpus	Nombre de mots	Vocabulaire : nombre de mots uniques	Fréquence des mots	Nombre de mots indexés dans WordNet
anglais	2 409 291 852	9 045 033	$\geq 600 = 96\,967$	47 118
français	804 476 834	4 348 227	$\geq 300 = 86\,697$	18 600
arabe	117 472 209	2 140 757	$\geq 300 = 33\,974$	1 646

TABLE 1 – Propriétés des corpus

6 Expérimentations et résultats

Pour chaque méthode, nous avons déterminé empiriquement les paramètres (taille de la fenêtre, dimension des vecteurs, nombre d'itérations, k le nombre de clusters, ...) qui donnent le meilleur résultat (Simwords maximum). Pour chaque corpus, nous avons utilisé le même nombre de clusters pour les trois méthodes de vectorisation. Les meilleurs résultats obtenus sont présentés dans le tableau 2 ci-dessous.

	Simwords					
	anglais		français		arabe	
	k = 2400	k = 1200	k = 800	k = 400	k = 300	k = 150
skip-gram	**0,603**	**0,542**	0,517	0,452	0,376	0,338
CBOW	0,581	0,522	**0,536**	**0,467**	**0,412**	**0,382**
GloVe	0,548	0,486	0,529	0,461	0,396	0,355
WebSOM	0,471	0,453	0,472	0,438	0,370	0,354

TABLE 2 – Évaluation des méthodes de représentations vectorielles de mots pour les trois langues : anglais, français et arabe, k est le nombre de clusters.

Il apparaît que word2vec, pour les deux modèles, représente mieux les mots selon la mesure proposée. skip-gram est plus performant pour le corpus anglais et CBOW pour le français et l'arabe. Nous ne prenons pas en compte les clusters singleton dans le calcul de Simwords, car ils augmentent les valeurs sans pour autant signifier une bonne méthode de vectorisation des mots. Par exemple, en segmentant les vecteurs produits par la méthode WebSOM avec le corpus français, on obtient plusieurs clusters singletons (429/800). Si nous les comptons, Simwords = 0,75522 au lieu 0.47218. En revanche, avec la méthode word2vec (CBOW), on obtient 166 clusters singleton sur 800 clusters, ce qui donne Simwords = 0,63301 au lieu de 0.53693 sans les compter. L'augmentation du nombre de clusters, augmente la valeur de Simwords (similarité entre mots) (voir la figure 2).

L'augmentation de la dimension des vecteurs pour word2vec et GloVe augmente les performances (figure 3, à gauche). L'augmentation de la taille de la fenêtre pour word2vec-CBOW, influence négativement les performances. Et pour GloVe, une fenêtre de 10 fait mieux que celles de 4 ou de 14 (figure 2, à droite). Les trois méthodes produisent quelques clusters parfaits (dont les mots appartiennent au même synset). Voici quelques exemples produits par CBOW avec Wikipédia en français : [façon, manière], [intersection, carrefour, croisement], [tremblement, séisme], [voyage, visite, séjour], [cour, tribunal], [organisme, corps, organe], [régler, résoudre], [résumé, synopsis].

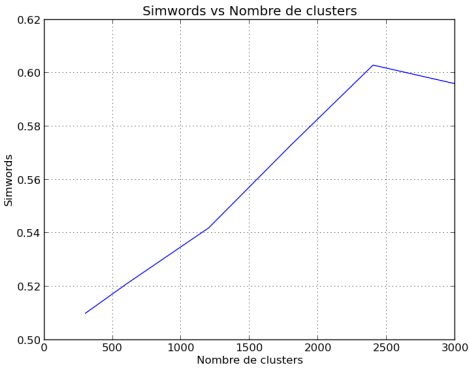


FIGURE 2 – Simwords vs Nombre de clusters : corpus anglais, avec word2vec (skip-gram), taille de la fenêtre = 8, dimension des vecteurs = 200.

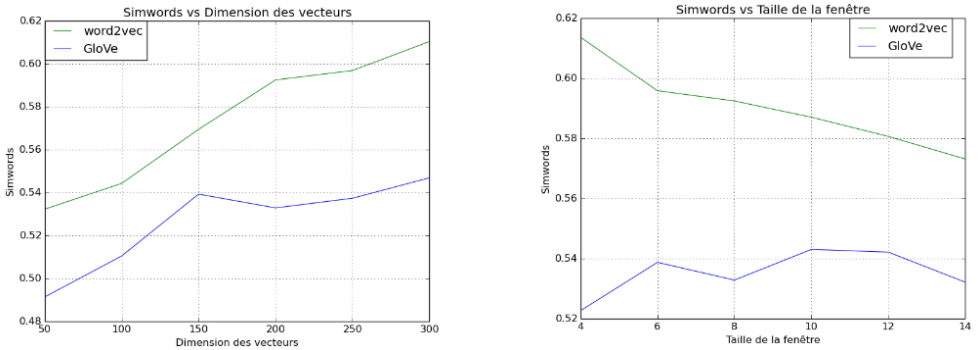


FIGURE 3 – À gauche, Simwords vs Dimension des vecteurs ; à droite, Simwords vs Taille de la fenêtre ; échantillon de 5000 vecteurs du corpus anglais, avec word2vec (CBOW) et GloVe, nombre de clusters = 800.

7 Conclusion

Nous avons proposé une solution d'évaluation de representations vectorielles de mots qui s'appuie sur un large lexique de référence. Elle permet de mesurer concrètement les performances d'une méthode de représentation vectorielle de mots, ou d'en optimiser les paramètres afin d'augmenter ses performances. Notre approche présente l'avantage d'être plus générale que les méthodes existant précédemment, grâce à la richesse sémantique qu'apportent les synonymes de WordNet et à la possibilité de l'utiliser avec différentes langues. Les 200 000 mots indexés dans WordNet offrent la possibilité de calculer à peu près 20 milliards de similarités entre mots. En revanche, avec "WordSim-353" par exemple, nous ne disposons que de 350 similarités (nombre de paires de mots). Ces résultats encourageants montrent l'efficacité de notre méthode et permettent d'envisager l'utilisation d'autres mesures de similarités dans WordNet et de comparer l'efficacité d'autres méthodes de vectorisation de mots.

Références

ARTHUR D. & VASSILVITSKII S. (2007). K-means++ : the advantages of careful seeding. In *In Proceedings of the 18th Annual ACM-SIAM Symposium on Discrete Algorithms*.

BARONI M., DINU G. & KRUSZEWSKI G. (2014). Don't count, predict ! a systematic comparison of context-counting vs. context-predicting semantic vectors. *52nd Annual Meeting of the Association for Computational Linguistics, ACL 2014 - Proceedings of the Conference*, **1**, 238–247.

CLAVEAU V. & KIJAK E. (2015). Thésaurus distributionnels pour la recherche d'information et vice-versa. In *Conférence en Recherche d'Information et Applications*, Actes de la conférence CORIA 2015, Paris, France.

FINKELSTEIN L., GABRILOVICH E., MATIAS Y., RIVLIN E., SOLAN Z., WOLFMAN G. & RUPPIN E. (2001). Placing search in context : The concept revisited. In *Proceedings of the 10th International Conference on World Wide Web*, WWW '01, p. 406–414, New York, NY, USA : ACM.

HARRIS Z. S. (1954). Distributional structure. *Word*, **10**, 146–162.

KOHONEN T., KASKI S., LAGUS K. & SALOJARVI J. (1998). Websom - self-organizing maps of document collection. *Helsinki University of Technologie, Finland*.

LANDAUER T. K. & DUTNAIS S. T. (1997). A solution to plato's problem : The latent semantic analysis theory of acquisition, induction, and representation of knowledge. *PSYCHOLOGICAL REVIEW*, **104**(2), 211–240.

LEVY O., GOLDBERG Y. & DAGAN I. (2015). Improving distributional similarity with lessons learned from word embeddings. *TACL*, **3**, 211–225.

MIKOLOV T., CHEN K., CORRADO G. & DEAN J. (2013). Efficient estimation of word representations in vector space. *arXiv preprint arXiv :1301.3781*.

MILLER G. A. (1995). Wordnet : A lexical database for english. *Commun. ACM*, **38**(11), 39–41.

PEDERSEN T., PATWARDHAN S. & MICHELIZZI J. (2004). Wordnet : :similarity : Measuring the relatedness of concepts. In *Demonstration Papers at HLT-NAACL 2004*, HLT-NAACL–Demonstrations '04, p. 38–41, Stroudsburg, PA, USA : Association for Computational Linguistics.

PENNINGTON J., SOCHER R. & MANNING C. D. (2014). Glove : Global vectors forword representation. In *EMNLP*, volume 14, p. 1532–1543.

WU Z. & PALMER M. (1994). Verbs semantics and lexical selection. In *Proceedings of the 32Nd Annual Meeting on Association for Computational Linguistics*, ACL '94, p. 133–138, Stroudsburg, PA, USA : Association for Computational Linguistics.

Traduction automatique de corpus en anglais annotés en sens pour la désambiguïsation lexicale d'une langue moins bien dotée, l'exemple de l'arabe

Marwa Hadj Salah[1, 2] Loïc Vial[1] Hervé Blanchon[1] Mounir Zrigui[2]
Benjamin Lecouteux[1] Didier Schwab[1]
(1) Univ. Grenoble Alpes, CNRS, Inria, Grenoble INP[*], LIG, 38000 Grenoble, France
(2) LaTICE, Tunis, 1008, Tunisie
Prénom.Nom@univ-grenoble-alpes.fr, Prénom.Nom@fsm.rnu.tn

RÉSUMÉ
Les corpus annotés en sens sont des ressources cruciales pour la tâche de désambiguïsation lexicale (*Word Sense Disambiguation*). La plupart des langues n'en possèdent pas ou trop peu pour pouvoir construire des systèmes robustes. Nous nous intéressons ici à la langue arabe et présentons 12 corpus annotés en sens, fabriqués automatiquement à partir de 12 corpus en langue anglaise. Nous évaluons la qualité de nos systèmes de désambiguïsation grâce à un corpus d'évaluation en arabe nouvellement disponible.

ABSTRACT
Automatic Translation of English Sense Annotated Corpora for Word Sense Disambiguation of a Less Well-endowed Language, the Example of Arabic

Sense-annotated corpus are decisive resources for Word Sense Disambiguation (WSD). Most of the languages have none or too little to build robust systems. In this article, we present 12 sense-annotated corpra for the Arabic language automatically build from 12 corpus in English. We evaluate the quality of our WSD systems using a newly available Arabic evaluation corpus.

MOTS-CLÉS : Désambiguïsation lexicale, Construction automatique de corpus annotés, .

KEYWORDS: Word Sense Disambiguation, Automatic translation of annotated corpus, .

1 Introduction

Les corpus annotés en sens sont des ressources cruciales pour la tâche de désambiguïsation lexicale (*Word Sense Disambiguation*). Cette tâche consiste à trouver pour chaque mot d'un texte le sens le plus approprié parmi un inventaire de sens pré-défini. Par exemple, dans la phrase « *Je vois la montagne à travers ma fenêtre.* », l'algorithme devrait choisir le sens du fenêtre qui correspond à la menuiserie plutôt que celui qui correspond à l'interface graphique.

Alors que l'anglais est la langue qui possède la plus grande quantité de telles ressources, la plupart des autres n'en possède pas ou trop peu pour pouvoir construire des systèmes robustes. Nous nous intéressons plus particulièrement ici à la langue arabe. Jusqu'en 2017, et la mise à disposition d'une version étendue de l'*OntoNote 5.0*, aucun corpus annoté manuellement en sens n'était librement disponible. Ce corpus, annoté avec des sens provenant du *Princeton WordNet* anglais pourrait devenir *de facto* le standard d'évaluation de la désambiguïsation lexicale (DL) de l'arabe. Toujours en 2017, notre équipe a également mis à disposition de la communauté la ressource UFSAC qui unifie un

∗. Institute of Engineering Univ. Grenoble Alpes

ensemble de 12 corpus existants en anglais (Vial *et al.*, 2017) annotés avec la version 3.0 du *Princeton WordNet*. Dans cet article, nous adaptons au format UFSAC et étendons la méthode introduite dans (Nasiruddin *et al.*, 2015) et (Hadj Salah *et al.*, 2016). Nous fabriquons ainsi de manière automatique 12 corpus en arabe que nous exploitons pour construire plusieurs systèmes de DL dont nous évaluons la qualité grâce au corpus *OntoNotes Release 5.0.*.

2 Contexte du travail

2.1 Désambiguïsation lexicale

Deux types de ressources sont nécessaires pour la DL : des bases lexicales et des corpus annotés en sens. Ce sont particulièrement les seconds qui sont absents pour la plupart des langues et en particulier pour l'arabe. Trois étapes sont nécessaires pour mettre en place une DL automatique (Schwab, 2017) : 1) *Constitution d'une ressource générique* : plusieurs ressources non dédiées à la DL sont possibles telles que les dictionnaires, les encyclopédies, les corpus non annotés, les corpus annotés, les bases lexicales etc. Certains de ces matériaux sont parfois construits automatiquement en utilisant d'autres matériaux. Cette étape est optionnelle et est souvent réalisée par des équipes spécialisées. 2) *Constitution d'une ressource dédiée à la DL* : on utilise une ou plusieurs ressources brutes pour donner une représentation informatique à chacun des sens d'un mot ; on constitue ici une ressource dédiée à la tâche. Les sens sont soit définis par l'expertise humaine, soit induits à partir des contextes d'utilisation dans les textes (induction de sens). Techniquement, la ressource peut-être un graphe, des définitions ou des représentations vectorielles. 3) *Utilisation de la ressource dédiée pour désambiguïser des textes* : il s'agit de l'algorithme de désambiguïsation proprement dit. Plusieurs facteurs peuvent entrer en compte. Certains sont communs à chaque algorithme comme la taille du contexte considéré pour le mot à désambiguïser (par exemple quelques mots avant ou après celui-ci, la phrase qui le contient, voire le texte) tandis que d'autres dépendent du type d'algorithme mis en œuvre : par exemple la limite à considérer pour la profondeur de la recherche dans un graphe ou encore les paramètres à prendre en compte pour des algorithmes stochastiques.

2.2 Désambiguïsation lexicale de l'arabe

Le fait que les diacritiques soient absents dans les textes arabes est la caractéristique la plus difficile pour la DL, car elle augmente le nombre de sens possibles d'un mot et rend la tâche de désambiguïsation plus difficile. Par ailleurs, la rareté ou la libre disponibilité de ressources (lexicales et/ou annotées) pour l'arabe complique non seulement la création de systèmes de DL pour cette langue mais empêche surtout la comparaison des systèmes entre eux. Pour avancer sur la désambiguïsation de l'arabe, il nous faut donc des corpus annotés en arabe pour apprendre un système de DL ainsi qu'un corpus annoté de référence pour faire de l'évaluation.

3 Méthode mise en oeuvre

Dans cette section nous présentons la méthode mise en œuvre afin de construire automatiquement des corpus annotés en sens dans la langue arabe. Pour ce faire, nous avons besoin de corpus parallèles bilingues afin de construire un système de traduction automatique, un système de DL supervisé, ainsi qu'un corpus de référence annoté en sens pour évaluer la désambiguïsation lexicale.

3.1 Prétraitement du Corpus annoté

Pour traduire un corpus à l'aide de notre système de traduction automatique statistique, celui-ci doit être normalisé pour être dans le même format que les données d'entraînement du système. Pour

ce faire, il est nécessaire d'éliminer les mots composés avec tiret bas existants dans le corpus, les mots non tokenisés, les mots commençant par une majuscule au début d'une phrase, etc. Cette normalisation se fait en trois étapes : 1) segmenter les mots composés (effacer le tiret bas) ; 2) appliquer la tokenisation Moses (ajouter des espaces entre mots et ponctuation) ; 3) mettre chaque mot du corpus dans une balise en suivant le format du corpus et en lui affectant un identifiant unique.

La Figure 1 présente un exemple de normalisation appliquée au mot composé "written_language" :

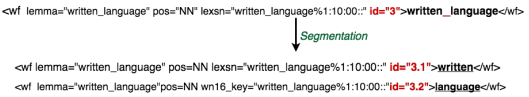


FIGURE 1 – Exemple de normalisation du mot composé "written_language"

3.2 Traduction et portage des annotations

Grâce à la boîte à outils Moses (Koehn *et al.*, 2007) et en exploitant l'ensemble des données parallèles alignées (LDC-Ummah, LDC-News, News Commentary, TED Talks), nous avons construit un système de traduction automatique statistique anglais-arabe afin de traduire de grands corpus annotés en sens et porter les annotations en source vers la langue cible. Nous avons sélectionné au hasard 800 lignes de chacun des corpus (3200 lignes au total) pour les données Test et Dev. Notre système a été évalué avec la métrique BLEU (score de 24,51%). Comme dans nos travaux sur l'amélioration de traduction vers le français d'un corpus annoté dans le contexte de la DL supervisée du français (Hadj Salah *et al.*, 2016) et précédemment dans (Nasiruddin *et al.*, 2015) où nous construisions deux corpus annotés en français et Bengali, nous nous servons des informations d'alignement des mots cible-source fournis par Moses (Koehn *et al.*, 2007) pour transférer les annotations d'un mot source anglais vers son correspondant dans la traduction arabe. (transfert d'annotation d'un mot source vers son correspondant dans la cible).

	LDC-Ummah	LDC-News	News Commentary	TED Talk
Nombre de mots arabes	2M	0.4M	3,9M	0.4M
Nombre de mots anglais	2.4M	0.5M	4.1M	0.5M

TABLE 1 – Description des corpus parallèles utilisés

3.3 Postraitement

Afin d'obtenir les meilleurs résultats possibles, après avoir traduit notre corpus de l'anglais vers l'arabe, nous mettons en œuvre des étapes de post-traitement pour corriger les problèmes (ordre et duplication de mots) posés par l'étape de portage des annotations qui repose sur les informations d'alignement fournies par Moses. Ainsi, nous avons développé un outil pour compiler une chaîne de post-traitement sur la sortie de traduction, qui suit les trois étapes suivantes : 1) Ré-ordonnancement et suppression des mots ajoutés par Moses ; 2) Concaténation des mots composés (ayant les mêmes id) afin d'obtenir un id unique pour chaque mot ; 3) Segmentation permettant d'avoir pour chaque mot, l'information grammaticale correspondante (POS), en utilisant l'analyseur morphologique MADAMIRA. De plus, étant donné que nous traitons de l'arabe comme langue cible, il est nécessaire de passer par une étape de détokensiation de la sortie de traduction pour avoir des mots arabes corrects.

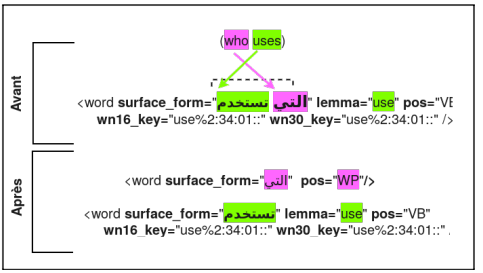


FIGURE 2 – Exemple de post-traitement

4 Application à la désambiguïsation lexicale

4.1 Corpus UFSAC

UFSAC (*Unification of Sense Annotated Corpora and Tools*) est une ressource récemment rendue disponible (Vial *et al.*, 2017). Elle regroupe l'ensemble des corpus en anglais annotés avec une version de *Princeton WordNet*. Ces corpus sont soit directement disponibles lorsque les droits le permettent, soit il est possible de les construire grâce au code source intégré à UFSAC à partir des données originales. UFSAC uniformise également ces corpus avec les sens du *Princeton WordNet* 3.0 (Miller, 1995). Dans les travaux décrits dans cet article, nous exploitons l'ensemble des 12 corpus d'UFSAC (voir tableau 2).

4.2 UFSAC-Arabe

Nous avons appliqué notre méthode pour créer des corpus UFSAC en arabe mais cette méthode pourrait être utilisée pour toute autre langue pourvu que l'on dispose d'un système de traduction de l'anglais vers cette langue.

Ressource	Phrases	Mots		Parties du discours annotées			
		Totaux	Annotés	Noms	Verbes	Adjectifs	Adverbes
SemCor	37176	767415	208142	80552	80079	29977	17534
DSO	101004	2494012	166436	99933	66503	0	0
WNGT	117659	1586199	456880	267985	69886	96299	22710
MASC	31760	548645	102614	45093	35283	11543	10695
OMSTI	820084	28455324	842999	437487	229976	175536	0
Ontonotes	124851	2331961	216283	75354	140929	0	0
SemEval 2007 Task 7	245	5589	1985	997	503	305	180
SemEval 2007 task 17	126	3012	380	135	245	0	0
SemEval 2013 task 12	306	7709	1439	1439	0	0	0
SemEval 2015 task 13	138	2677	959	504	235	144	76
SensEval 2	238	5741	2063	973	492	364	234
SensEval 3	300	5493	1738	806	640	281	11

TABLE 2 – Chiffres relatifs à notre ensemble de corpus en langue arabe annotés en sens

5 Évaluation

Pour évaluer la pertience de notre approche, nous utilisons un système de désambiguisation lexicale supervisée classique en ré-implantant de la méthode NUP-PT utilisée sur l'anglais lors de la compétition SemEval 2007.

5.1 Système de désambiguïsation basé sur les machines à vecteurs de support

L'apprentissage automatique consiste à entraîner un classfieur pour chaque mot cible dans le but de prédire le sens le plus pertinent dans son contexte. Les algorithmes supervisés utilisent des techniques d'apprentissage automatique. Ils apprennent un classifieur sur les corpus annotés en sens en utilisant des classifieurs classiques : séparateurs à vaste marge (NUS-PT (Chan *et al.*, 2007)), classifieurs naïfs bayésiens (NUS-ML, (Cai *et al.*, 2007)), combinaison de séparateurs à vaste marge, entropie maximale (LCC-WSD, (Novischi *et al.*, 2007)). On ne peut pas vraiment affirmer que tel ou tel classifieur soit meilleur qu'un autre et ce qui différencie les performances des systèmes est principalement et directement lié à la taille des données annotées.

Pour cet article, nous avons ré-implanté le classifieur utilisé dans le système NUS-PT qui était le système supervisé état de l'art avant l'émergence des réseaux de neurones profonds. Nous avons fait ce choix pour deux raisons : 1) Prouver la pertience de l'approche ; 2) Utiliser un système de calcul moins gourmand en ressource et donc accessible à un plus grand nombre de chercheurs.

Notre classifieur se base sur trois ensembles de traits pour assigner un sens à un mot donné : 1) Les parties du discours des mots voisins (P_i), 7 traits sont extraits, qui correspondent aux labels de partie du discours des trois mots à gauche (P_{-3}, P_{-2}, P_{-1}) trois mots à droite (P_1, P_2, P_3) et à celui du mot cible (P_0) ; 2) Les collocations locales ($C_{i,j}$), qui correspondent à la suite ordonnée des mots entre les index i et j relativement au mot cible, mis en lettres minuscules. 11 traits sont ainsi extraits : $C_{-1,-1}$, $C_{1,1}$, $C_{-2,-2}$, $C_{2,2}$, $C_{-2,-1}$, $C_{-1,1}$, $C_{1,2}$, $C_{-3,-1}$, $C_{-2,1}$, $C_{-1,2}$ et $C_{1,3}$; 3) Le contexte voisin, ce trait correspond à un vecteur de la taille du nombre de lemmes différents observés pendant l'entraînement. Chaque composante du vecteur correspond ainsi à un lemme, et sa valeur est mise à 1 si le lemme d'un des mots présents dans la même phrase que le mot cible correspond au lemme de cette composante. Elle vaut 0 sinon.

5.2 Corpus d'évaluation : OntoNotes Release 5.0

OntoNotes Release 5.0 (Weischedel *et al.*, 2015) comporte trois langues (anglais, arabe et chinois). C'est un grand corpus annoté manuellement en sens libre de droit contenant plusieurs genres de textes (News, conversations téléphoniques, weblogs, usenet newsgroups, broadcast, talk shows) pour l'anglais et le chinois et seulement des données News pour la partie arabe, avec des informations structurelles (syntaxe et structures prédicat-arguments) et sémantiques superficielles (sens du mot lié à une ontologie et coréférence). La partie arabe d'*OntoNotes Release 5.0* comprend 300K mots du corpus arabe *An-Nahar Newswire*. C'est sur cette partie que nous évaluons notre système tandis que nous l'entraînons en partie sur les traductions de la partie anglaise. Il n'y a pas de biais car les parties arabes et anglaises ne sont pas des traductions l'une de l'autre. Le tableau ci dessous présente le nombre de lemmes, de sens ainsi que les mapping$_{WordNet}$ pour les verbes et noms en arabe.

	#Lemmes	#Lemmes uniques	#Sens uniques	**#Correspondances$_{WordNet}$ uniques**
Verbes	3990	150	642	**4182**
Noms	8534	111	463	**1376**
Total	12524	261	1105	**5558**

TABLE 3 – Description de la partie arabe annotée en sens d'OntoNotes Release 5.0

5.3 Mesures d'évaluation

Nous évaluons notre système de DL *in vitro*, en exploitant le corpus annoté de référence cité précédemment, et en utilisant les mesures d'évaluation classiques comme la plupart des tâches de DL telle

que *SemEval 2013* : en termes de précision P, de rappel R et du score F1 qui correspond à la moyenne harmonique de P et R. La précision se définit comme :

$$P = \frac{\text{nombre de mots annotés correctement}}{\text{nombre de mots annotés}} \quad R = \frac{\text{nombre de mots annotés correctement}}{\text{nombre de mots à annoter}} \quad F1 = \frac{2 \cdot P \cdot R}{P+R}$$

5.4 Résultats et analyse

Dans cette partie, nous présentons les résultats de notre système de DL sur l'anglais afin de montrer qu'il obtient des résultats état de l'art hors réseaux de neurones puis sur l'arabe pour montrer que nous obtenons de bons résultats sur nos corpus automatiquement générés pour l'arabe.

5.4.1 Résultats de DL sur l'anglais

Afin de vérifier l'efficacité de notre méthode et ainsi évaluer l'influence de l'étape de traduction et le portage des annotations sur la qualité de la DL arabe, nous avons entraîné un système de désambiguisation lexicale SVM anglais nommé SVM-UFSAC-eng sur les corpus UFSAC originaux en anglais (Semcor, DSO, WNGT, MASC, OMSTI et OntoNotes anglais) comportant 1.99M mots annotés. Ensuite, ce système a été évalué sur différents corpus ($SensEval2$, $SensEval3task1$ et $semeval2007task17$) pour comparer les résultats.

Système	Tâche	Précision	Rappel	Score F1
SVM-UFSAC-eng	SensEval2	71.34	69.57	70.45
SVM-UFSAC-eng +repli premier sens	SensEval2	71.88	71.88	**71.88**
IMS+emb	SensEval2	68,3	68,3	68,3
SVM-UFSAC-eng	SensEval3task1	65.32	59.58	62.31
SVM-UFSAC-eng+repli premier sens	SensEval3task1	65,50	65.50	**65.50**
IMS+emb	SensEval3task1	68.2	68.2	68.2
SVM-UFSAC-eng	semeval2007task17	60.92	60.65	60.79
SVM-UFSAC-eng+repli premier sens	semeval2007task17	60.87	60.87	**60.87**
IMS+emb	semeval2007task17	68.2	68.2	59.7

TABLE 4 – Performances de notre système de désambiguïsation lexicale sur l'anglais

D'après le tableau ci-dessous, nous remarquons que le système SVM-UFSAC-eng a obtenu les meilleurs performances en termes de score F1 sur les corpus $SensEval2$, $SensEval3task1$ et $semeval2007task17$, respectivement 71.88%, 65.50% et 60.87%, en ajoutant le repli vers le premier sens.

D'autre part, nous avons reporté dans le tableau les résultats du meilleur système de désambiguïsation supervisé état de l'art (Iacobacci *et al.*, 2016) (nommé IMS+emb). Nous observons que les performances de SVM-UFSAC-eng sont comparables.

Dans ce qui suit, nous présentons nos expériences sur la langue arabe, nous montrons que nos résultats sont convenables et encourageants comparés à l'anglais.

5.4.2 Résultats de DL sur l'arabe

Nous avons réalisé l'entraînement de notre algorithme de DL SVM (voir section 5.1) sur les douze corpus traduits depuis l'anglais comportant 2M de mots annotés. Nous appelons ce système SVM-UFSAC-ara.

Le tableau 5 présente les résultats de notre système. Comme beaucoup d'algorithmes de DL, SVM-UFSAC-ara n'annote pas l'ensemble des termes. Par exemple, si les corpus annotés ne contiennent pas d'exemples pour un mot à étiqueter, il ne peut pas réaliser cette opération. Nous utilisons alors l'heuristique classique qui consiste à choisir le premier sens de *Princeton WordNet*.

	Précision	Rappel	Score F1
SVM-UFSAC-ara	68.60	62.14	65.21
SVM-UFSAC-ara+ repli premier sens	67.55	62.74	65.06
SVM-UFSAC-ara+Post-traitement	**70.86**	64.20	**67.36**
SVM-UFSAC-ara+Post-traitement + repli premier sens	69.75	**64.79**	67.18

TABLE 5 – Performances de notre système de désambiguïsation lexicale arabe

Ces résultats montrent qu'il est possible de créer des systèmes de DL pour des langues peu dotées comme l'arabe, qui n'ont pas (ou trop peu) de données annotées en sens. À notre connaissance, il s'agit du premier système évalué sur le corpus OntoNotes, il n'est donc pas possible de se comparer à d'autres systèmes. Toutefois, il convient de noter que nous obtenons des résultats similaires aux résultats obtenus habituellement sur des tâches d'évaluation de l'anglais.

Notre système de désambiguïsation arabe a été évalué en termes de *Précision, Rappel* et *Score F1* sur le corpus de référence OntoNotes arabe. Notre système a réussi à désambiguïser 11346 mots annotés sur 12524 ; les mots non annotés n'étant pas présent dans le corpus d'entraînement.

En lisant le tableau 5, nous pouvons remarquer tout d'abord, qu'en ajoutant le repli vers le premier sens pour les mots non annotés, le système SVM-UFSAC a obtenu 67.55% en termes de précision et 65.06% en termes du score F1. En outre, en appliquant les différentes étapes de post-traitement décrites précédemment sur les données traduites en arabe, notre système de désambiguïsation obtient une meilleure performance en termes de précision 70.86% (+3.31%) , c'est-à-dire qu'il a été capable de désambiguïser correctement 8040 mots parmi les 11346 mots annotés, et en termes du score F1 67.36% (+2.30%).

Par conséquent, on peut dire que nous avons obtenu des résultats de DL arabe similaires aux résultats obtenus pour l'anglais sachant que les corpus d'entraînement utilisés pour les deux langues ont presque la même taille, ce qui prouve l'efficacité de notre méthode.

6 Conclusion et perspectives

Dans cet article, nous avons exposé les difficultés posées par la langue arabe lors de la création et l'évaluation des systèmes de désambiguïsation lexicale. Nous avons montré que l'absence de corpus annotés en sens en est la cause. Pour palier ce manque de ressources, nous proposons à la communauté jusqu'à 12 corpus arabes annotés en sens. Ces corpus sont obtenus par traduction automatique et portage d'annotations de corpus anglais annotés en sens. Ces corpus peuvent par exemple être utilisés pour l'apprentissage de systèmes de désambiguïsation lexicale. Nous avons exploité des corpus annotés pour certaines langues, ici l'anglais, pour créer rapidement un système de désambiguisation lexicale supervisée pour une langue moins dotée telle que l'arabe. Nos résultats prouvent la pertinence de notre approche et sont très encourageants. Il est donc possible de fabriquer des systèmes de désambiguïsation lexicale de bonne qualité pour n'importe quelle langue dès lors que l'on dispose d'un système de traduction automatique de bonne qualité de l'anglais vers cette langue. L'ensemble des douze corpus en arabe et les scripts permettant de les réaliser seront disponibles pour la communauté.

Références

F. BENARMARA, N. HATOUT, P. MULLER & S. OZDOWSKA, Eds. (2007). *Actes de TALN 2007 (Traitement automatique des langues naturelles)*, Toulouse. ATALA, IRIT.

CAI J. F., LEE W. S. & TEH Y. W. (2007). Nus-ml : Improving word sense disambiguation using topic features. In *Proceedings of the 4th International Workshop on Semantic Evaluations*, SemEval '07, p. 249–252, Stroudsburg, PA, USA : Association for Computational Linguistics.

CHAN Y. S., NG H. T. & ZHONG Z. (2007). Nus-pt : exploiting parallel texts for word sense disambiguation in the english all-words tasks. In *Proceedings of the 4th International Workshop on Semantic Evaluations*, p. 253–256 : Association for Computational Linguistics.

G. DIAS, Ed. (2015). *Actes de TALN 2015 (Traitement automatique des langues naturelles)*, Caen. ATALA, HULTECH.

HADJ SALAH M., BLANCHON H., ZRIGUI M. & SCHWAB D. (2016). Amélioration de la traduction automatique d'un corpus annoté. In *JEP-TALN-RECITAL 2016*.

IACOBACCI I., PILEHVAR M. T. & NAVIGLI R. (2016). Embeddings for word sense disambiguation : An evaluation study. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1 : Long Papers)*, p. 897–907, Berlin, Germany : Association for Computational Linguistics.

KOEHN P., HOANG H., BIRCH A., CALLISON-BURCH C., FEDERICO M., BERTOLDI N., COWAN B., SHEN W., MORAN C., ZENS R. *et al.* (2007). Moses : Open source toolkit for statistical machine translation. In *Proceedings of the 45th annual meeting of the ACL on interactive poster and demonstration sessions*, p. 177–180 : Association for Computational Linguistics.

LAIGNELET M. & RIOULT F. (2009). Repérer automatiquement les segments obsolescents à l'aide d'indices sémantiques et discursifs. In *Actes de TALN 2009 (Traitement automatique des langues naturelles)*, Senlis : ATALA LIPN.

LANGLAIS P. & PATRY A. (2007). Enrichissement d'un lexique bilingue par analogie. In (Benarmara *et al.*, 2007), p. 101–110.

MILLER G. A. (1995). Wordnet : a lexical database for english. *Communications of the ACM*, **38**(11), 39–41.

NASIRUDDIN M., TCHECHMEDJIEV A., BLANCHON H. & SCHWAB D. (2015). Création rapide et efficace d'un système de désambiguïsation lexicale pour une langue peu dotée. In *TALN 2015-22ème Conférence sur le Traitement Automatique des Langues Naturelles,, Caen, France*.

NOVISCHI A., SRIKANTH M. & BENNETT A. (2007). Lcc-wsd : System description for english coarse grained all words task at semeval 2007. In *Proceedings of the 4th International Workshop on Semantic Evaluations*, SemEval '07, p. 223–226, Stroudsburg, PA, USA : Association for Computational Linguistics.

SCHWAB D. (2017). Cours master mosig.

SERETAN V. & WEHRLI E. (2007). Collocation translation based on sentence alignment and parsing. In (Benarmara *et al.*, 2007), p. 401–410.

VIAL L., LECOUTEUX B. & SCHWAB D. (2017). Uniformisation de corpus anglais annotés en sens. In *24ème Conférence sur le Traitement Automatique des Langues Naturelles*, Orléans, France.

WEISCHEDEL R., PALMER M., MARCUS M., HOVY E., PRADHAN S., RAMSHAW L., XUE N., TAYLOR A., KAUFMAN J., FRANCHINI M., EL-BACHOUTI M., BELVIN R. & HOUSTON A. (2015). Ontonotes release 5.0. *LDC2013T19. Web Download. Philadelphia : Linguistic Data Consortium*.

Détection de mésusages de médicaments dans les réseaux sociaux

Elise Bigeard[1, 2] Natalia Grabar[1] Frantz Thiessard[2,3]
(1) CNRS, Univ Lille, UMR 8163 STL - Savoirs Textes Langage, F-59000 Lille, France
(2) Univ. Bordeaux, Inserm, Bordeaux Population Health Research Center, team ERIAS, UMR 1219, F-33000 Bordeaux, France
(3) CHU de Bordeaux, Pole de sante publique, Service d'information medicale, F-33000 Bordeaux, France
elise.bigeard@u-bordeaux.fr

RÉSUMÉ

Un mésusage apparaît lorsqu'un patient ne respecte pas sa prescription et fait des actions pouvant mener à des effets nocifs. Bien que ces situations soient dangereuses, les patients ne signalent généralement pas les mésusages à leurs médecins. Il est donc nécessaire d'étudier d'autres sources d'information pour découvrir ce qui se passe en réalité. Nous proposons d'étudier les forums de santé en ligne. L'objectif de notre travail consiste à explorer les forums de santé avec des méthodes de classification supervisée afin d'identifier les messages contenant un mésusage de médicament. Notre méthode permet de détecter les mésusages avec une F-mesure allant jusqu'à 0,810. Cette méthode peut aider dans la détection de mésusages et la construction d'un corpus exploitable par les experts pour étudier les types de mésusages commis par les patients.

ABSTRACT

Detection of drug misuse in social media.

Misuses happen when patients do not follow the prescriptions and do actions which lead to potentially harmful situations. Although such situations are dangerous, patients usually do not report the misuse of drugs to their physicians. Hence, it is necessary to study other sources of information to discover what is happening in reality. We propose to study online health fora. The purpose of our work is to explore online health fora with supervised classification methods in order to identify messages that contain drug misuses. Our method permits to detect misuses with up to 0.810 F-mesure. This method can help the detection of messages with misuses and building corpus with drug misuses, which can be used by experts to further study the types of misuses committed by patients.

MOTS-CLÉS : TAL, forums de discussion en santé, classification supervisée, domaine médical, mésusage de médicaments.

KEYWORDS: NLP, health discussion forums, supevised classification, medical domain, drug misuse.

1 Introduction

Un mésusage de médicament apparaît lorsqu'un patient ne respecte pas sa prescription : sous-dosage, sur-dosage, utilisation de médicaments pour des raisons autres de celles de la prescription, consommation de médicaments prescrits pour une autre personne, etc. Ces situations sont dangereuses car elles mettent en danger la santé de la personne. Comme les patients reportent rarement les mésusages à leurs médecins, il est nécessaire de consulter d'autres sources d'informations pour

découvrir ce qui se passe en réalité. Nous proposons d'étudier les réseaux sociaux, où les patients communiquent librement et facilement sur leur processus de santé (Gauducheau, 2008), et sans doute sur l'usage de médicaments. Actuellement, les réseaux sociaux sont largement étudiés par plusieurs disciplines et en poursuivant différents objectifs : identification de géolocalisation, fouille d'opinions, extraction d'événements, traduction et résumé automatique... (Louis, 2016). Dans le domaine médical, qui est au centre de notre travail, les réseaux sociaux peuvent être exploités pour fournir des informations pour la surveillance épidémiologique (Collier, 2011; Lejeune *et al.*, 2013), la qualité de vie des patients (Tapi Nzali, 2017) et les effets indésirables de médicaments (Morlane-Hondère *et al.*, 2016). Cependant, peu de travaux s'intéressent au mésusage de médicaments. Nous pouvons citer par exemple l'étude des tweets concernant l'usage non-médical de médicaments avec des méthodes non supervisées (Kalyanam *et al.*, 2017) et la création d'une plateforme générique pour l'étude de sur-usages (Cameron *et al.*, 2013).

L'objectif général de notre travail consiste également à aider l'étude des mésusages commis par les patients. Notre travail propose d'identifier, au sein des forums de discussion, les messages contenant un mésusage. Dans la suite de ce travail, nous présentons d'abord le matériel (section 2) et les étapes de la méthode (section 3). Nous présentons et discutons les résultats ensuite (section 4).

2 Matériel

Nous utilisons plusieurs types de matériel décrit dans cette section : un corpus de messages postés sur des forums de santé, un lexique de médicaments et un lexique de maladies.

Corpus. Nous construisons le corpus à partir de deux forums de Doctissimo : médicaments [1] et grossesse [2]. Les messages collectés ont été postés entre 2010 et 2015. Nous conservons uniquement les messages contenant au moins un nom de médicament et excluons les messages de plus de 2 500 caractères, dont le contenu est hétérogène et difficile à analyser. Le corpus obtenu contient 119 562 messages (15,5M mots).

Noms de médicaments. Nous utilisons un ensemble de noms commerciaux de médicaments associés à leur code ATC (Skrbo *et al.*, 2004) provenant de différentes sources : la base CNHIM Thériaque [3], la *base publique du médicament* [4] et la base *Medic'AM* de l'Assurance Maladie [5]. Ce lexique contient 4 133 noms de médicaments répartis en 1 690 codes ATC distincts.

Noms de maladies. Nous utilisons un ensemble de 29 maladies traitées par les médicaments anxiolytiques et antidépresseurs. Les maladies sont associées à leurs codes CIM-10 (OMS, 1995) : *panique/F41.0*, *anxiété/F41.9*... Ce codage, employé par les professionnels de santé, fait une distinction fine des maladies, qui reste difficile à faire pour les patients. En effet, les non-spécialistes peuvent confondre les maladies, comme par exemple *agoraphobie/F40.0* et *phobie sociale/F40.1*, *panique/F41.0* et *anxiété/F41.9*. Un groupement simplifié est donc effectué par les experts. Ce lexique source est étendu dans une expérience antérieure grâce à l'utilisation de lexiques existants et à l'analyse du corpus (Bigeard, 2017).

1. http://forum.doctissimo.fr/medicaments/liste_categorie.htm
2. http://forum.doctissimo.fr/grossesse-bebe/liste_categorie.htm
3. http://www.theriaque.org
4. http://base-donnees-publique.medicaments.gouv.fr
5. https://www.ameli.fr/l-assurance-maladie/statistiques-et-publications/donnees-statistiques/medicament/medic-am/medic-am-mensuel-2017.php

3 Méthodes

3.1 Prétraitements

Les messages sont segmentés en mots, étiquetés et lemmatisés avec Treetagger (Schmid, 1994). La casse est neutralisée et les diacritiques sont supprimées pour diminuer la variation orthographique, comme {*Anxiété*, *anxiete*}. Aucune correction orthographique n'est effectuée. Les mots grammaticaux (articles, prépositions, verbes auxiliaires...) sont supprimés.

Nous avons effectué des expériences avec plusieurs formes du texte :

— texte non lemmatisé,
— texte lemmatisé, chiffres remplacés par la séquence *nombre*,
— texte lemmatisé, mots grammaticaux supprimés.

Les messages sont annotés avec les lexiques de médicaments et de maladies, et indexés selon le code correspondant : CIM-10 pour les maladies et ATC pour les médicaments. Nous pouvons ainsi remarquer que certaines classes de médicaments sont très fréquentes, avec 60 % des messages dédiés à la pilule contraceptive et 15 % aux antidépresseurs et anxiolytiques.

3.2 Annotation manuelle et Corpus de référence

Afin de constituer un corpus d'entraînement et de test, une annotation manuelle est réalisée. Trois annotateurs ont pour objectif d'associer chaque message à l'une des catégories suivantes :

usage normal : contient un usage normal de médicament, comme dans : *Mais la question que je pose est 'est ce que c'est normal que le loxapac que je prends met des heures à agir ? ? ?*

pas d'usage : ne contient pas d'usage de médicament, comme dans : *ouf boo, repose toi surtout, il ne t'a pas prescris d'aspegic nourisson ? ?*

mésusage : contient un mésusage. Lorsque cette catégorie est sélectionnée, l'annotateur explique brièvement en quoi consiste le mésusage (sur-dosage, sevrage brutal...). Dans l'exemple qui suit, le mésusage est dû à un oubli de prise : *bon moi la miss boulette et la tete en l'air je devais commencer mon "utrogestran 200" a j16 bien sur j'ai oublier ! donc je l'ai pris ce soir ! ! ! !*

incertitude : impossible de décider

Trois annotateurs participent à la tâche : un pharmacologue et deux informaticiens familiers avec les textes médicaux. Deux annotateurs (un pharmacologue et un informaticien) effectuent l'annotation, alors que le troisième vérifie le contenu de la catégorie de mésusage. Les cas de désaccord ou de messages annotés *mésusage* sont discutés entre les annotateurs pour trouver la catégorie consensuelle.

Les données de référence sont constituées à partir de trois corpus :

— $C1$ contient 150 messages sélectionnés aléatoirement. Chaque message est annoté indépendamment par deux annotateurs. Nous utilisons cette annotation pour calculer l'accord inter-annotateur selon la mesure du Kappa (Cohen, 1960). En cas de désaccord, les annotateurs se concertent et décident ensemble de la catégorie finale ;
— $C2$ contient 1 200 messages sélectionnés aléatoirement. Il est divisé en deux et chaque partie est annotée par l'un des deux annotateurs ;
— $C3$ contient 500 messages. Puisque certaines classes de médicaments sont plus fréquentes, nous construisons $C3$ pour qu'il contienne des médicaments plus variés : pour chacune des

50 classes les plus fréquentes dans le corpus, 10 messages sont sélectionnés aléatoirement. Ce corpus permet de diversifier le contenu étudié. Il est annoté par l'expert en pharmacologie.

Le corpus total contient 1 850 messages (202 726 mots) dont 600 dans la catégorie *pas d'usage*, 1 117 dans la catégorie *usage normal* et 133 dans la catégorie *mésusage*.

3.3 Catégorisation supervisée

Nous utilisons l'implémentation Weka (Witten & Frank, 2005) de divers algorithmes de catégorisation supervisée : NaiveBayes (John & Langley, 1995), Bayes Multinomial (McCallum & Nigam, 1998), J48 (Quinlan, 1993), Random Forest (Breiman, 2001) et Simple Logistic (Landwehr *et al.*, 2005). Nous utilisons trois types de descripteurs : texte lemmatisé et vectorisé, classes ATC des médicaments, classes CIM-10 des maladies.

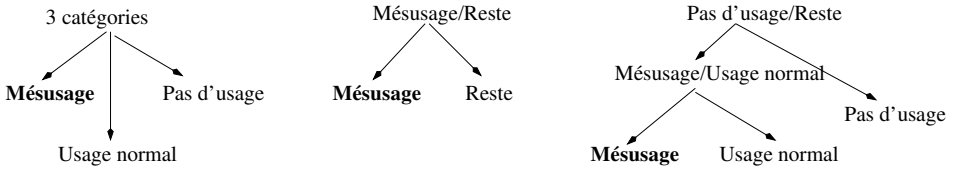


FIGURE 1 – Organisation des expériences pour la détection des mésusages

L'objectif des expériences est d'identifier les messages contenant un mésusage de médicament. Trois chemins sont possibles pour y arriver (figure 1) :

— *Trois catégories.* Nous utilisons directement les trois catégories de l'annotation. Chaque catégorie contient 133 messages. Cette distinction est sans doute la plus difficile car le modèle doit différencier les trois catégories en même temps ;
— *Catégorisation binaire mésusage-reste.* Ce modèle contraste la catégorie *mésusage* avec les deux autres (*usage normal* et *pas d'usage*). Le corpus contient 133 messages dans la catégorie *mésusage* et 133 messages dans les deux autres catégories. Ce modèle est la méthode la plus directe pour détecter les mésusages ;
— *Catégorisation binaire pas d'usage-reste* suivie de *catégorisation binaire usage normal-mésusage*. Cette expérience est effectuée en deux temps : il s'agit d'abord d'isoler les cas de *pas d'usage* (le corpus contient alors 2*300 messages) et ensuite de distinguer entre les cas de *mésusage* et d'*usage normal*. Il s'agit d'un modèle en cascade. En raison du faible nombre de messages de type *mésusage*, lors de l'évaluation de la seconde étape, nous exploitons l'ensemble des messages de types *mésusage*, ce qui nous fournit un corpus de 2*133 messages.

Nous effectuons quatre expériences principales, où nous utilisons les descripteurs suivants :
— *texte* : le texte lemmatisé et vectorisé uniquement ;
— *médicaments* : le texte lemmatisé et vectorisé, avec les codes de médicaments ;
— *maladies* : le texte lemmatisé et vectorisé, avec les codes de maladies ;
— *médicaments+maladies* : le texte lemmatisé et vectorisé, avec les codes de médicaments et de maladies.

Ces descripteurs permettront d'observer l'impact des types de médicaments et de maladies par rapport au seul texte des messages. Pour mieux comprendre le rôle de médicaments et de maladies,

TABLE 1 – Résultats pour l'expérience mésusage / reste avec le descripteur médicaments

	NaiveBayes			NaiveBayes Multinomial		
	Précision	Rappel	F-mesure	Précision	Rappel	F-mesure
formes	0,735	0,734	0,733	0,669	0,646	0,627
lemmes	0,778	0,772	0,773	0,608	0,582	0,579
lemmes lexicaux	0,821	0,810	0,809	0,812	0,810	**0,810**

TABLE 2 – Résultats pour l'expérience mésusage / usage avec le descripteur médicaments

	NaiveBayes			NaiveBayes Multinomial		
	Précision	Rappel	F-mesure	Précision	Rappel	F-mesure
formes	0,733	0,734	0,734	0,641	0,646	0,633
lemmes	0,782	0,772	0,772	0,787	0,759	0,758
lemmes lexicaux	0,801	0,797	**0,797**	0,661	0,595	0,560

nous effectuons deux ensembles d'expériences supplémentaires, pour les médicaments et pour les maladies :

- *normal :* le texte des messages lemmatisé et vectorisé ;
- *code :* les noms de médicaments ou de maladies sont remplacés par leurs codes ATC ou CIM-10 ;
- *normal+code :* le texte des messages contient les noms de médicaments et maladies, et leurs codes sont ajoutés en plus ;
- *substitution :* les noms de médicaments et de maladies sont remplacés par la séquence *drug* ou *disorder* ;
- *supprimé :* les noms de médicaments et de maladies sont supprimés.

Nous ne disposons que de 133 messages dans la catégorie *mésusage*, qui est aussi la catégorie que nous cherchons à isoler. Pour chaque expérience, le nombre de messages de chaque catégorie est équilibré afin que les classifieurs accordent de l'importance à la catégorie *mésusage*. Les messages sont sélectionnés aléatoirement. Un même corpus est utilisé pour chaque séquence d'expériences. 70 % du corpus est utilisé pour l'entraînement et 30 % pour l'évaluation. Nous ne faisons donc pas de cross-validation. Les résultats sont évalués avec des mesures classiques (Sebastiani, 2002) : vrais positifs pour les messages correctement classifiés ; faux négatifs pour les messages non détectés ; faux positifs pour les messages détectés à tord ; de même que la précision P, le rappel R et la F-mesure F.

4 Résultats et Discussion

L'accord inter-annotateur est de 0,46, ce qui est un accord modéré (Landis & Koch, 1977) et indique qu'il s'agit d'une tâche potentiellement difficile. En effet, il existe de nombreux cas où il est difficile de décider si le mésusage a lieu ou pas : par exemple, lorsque le patient exprime une intention de commettre un mésusage ou un questionnement qui peut mener à un mésusage selon la réponse obtenue. Dans cet exemple *"mon psy m'a prescrit effexor enfin son generique a 37.5 je le prend ou pas ? jai trop peur des effet secondaire"* si la personne décide de ne pas prendre un médicament qui lui a été prescrit elle se retrouve en situation de mésusage, mais l'usage restera normal si la personne suit la prescription.

Les expériences de détection automatique de mésusages montrent que la *catégorisation binaire mésusage-reste* atteint le meilleur résultat avec 0,810 de F-mesure (l'algorithme NaiveBayes Multinomial, le texte lemmatisé, sans les mots grammaticaux et avec les descripteurs *médicaments*), comme présenté dans le tableau 1. La catégorisation en cascade obtient 0,756 de F-mesure pour la tâche *pas d'usage / reste* (l'algorithme NaiveBayes, le texte lemmatisé, sans les mots grammaticaux, avec les descripteurs *médicaments*) et 0,797 de F-mesure pour la tâche *usage / mésusage* (les mêmes paramètres), comme indiqué dans le tableau 2. Le modèle *trois catégories* atteint au mieux 0,607 de F-mesure (l'algorithme NaiveBayes Multinomial, le texte lemmatisé, les descripteurs *médicaments*).

TABLE 3 – Résultats de l'expérience sur les médicaments, avec texte lemmatisé, algorithme de type NaiveBayes, exprimés en F-mesure

descripteurs	3 classes	mésusage/reste	pas d'usage/reste	usage/mésusage
normal	0,544	0,734	**0,713**	0,741
code	0,546	0,728	0,700	0,739
normal + code	**0,563**	**0,740**	0,684	**0,758**
substitution	0,540	0,731	0,702	0,721
supprimé	0,554	0,731	0,694	0,721

TABLE 4 – Résultats de l'expérience sur les maladies, avec texte lemmatisé, algorithme de type NaiveBayes, exprimés en F-mesure

descripteurs	3 classes	mésusage/reste	pas d'usage/reste	usage/mésusage
normal	0,577	0,763	**0,768**	**0,763**
code	0,542	0,720	0,750	0,651
normal + code	**0,579**	**0,793**	0,749	**0,763**
substitution	0,554	0,734	0,755	0,661
supprimé	0,544	0,730	0,751	0,676

Les résultats des expériences sur les médicaments et les maladies sont présentés dans les tableaux 3 et 4. Pour chaque expérience, la F-mesure obtenue avec le meilleur algorithme (NaiveBayes ou NaiveBayes Multinomial) est présentée. Le meilleur résultat pour chaque tâche est mis en valeur en gras, ce qui nous permet de constater que le descripteur *normal* ou *normal + code* dépasse toujours les autres ensembles de descripteurs. Il semble donc que le nom même du médicament ou de la maladie soient exploités par les classifieurs, plutôt que leur catégorie (descripteur *code*) ou leur simple présence (descripteur *substitution*). Cependant, on note que les résultats obtenus par les différents descripteurs sont proches les uns des autres, ce qui suggère que les noms des médicaments et des maladies ne sont que peu exploités par les classifieurs.

Notons de plus que (1) dans la majorité des expériences, les algorithmes bayésiens se montrent les plus efficaces, aussi avons-nous détaillé uniquement les résultats obtenus avec ces algorithmes ; (2) les noms de médicaments et de maladies ont un rôle positif sur la détection de mésusages ; (3) la lemmatisation et la suppression de mots grammaticaux ont un effet positif ; (4) la précision est en général plus élevée que le rappel, mais l'écart entre les deux reste faible.

Une analyse des faux positifs et des faux négatifs permet de faire les observations suivantes :

— Pour l'expérience *pas d'usage/reste*, 27 messages sont incorrectement classifiées dans *reste* et 33 dans *pas d'usage*. Parmi ces 33 messages, 11 ne contiennent pas d'information explicite sur la prise du médicament, comme dans *elina a quoi pour sa toux ? Ici antibio rebelotte.*

Dans 5 messages, le médicament n'est pas cité mais reste sous-entendu à travers la maladie, comme dans *j'ai pris mon traitement et les allergies ça va mieux et aussi un spray nasal* ;

— Pour l'expérience *mésusage/reste*, 12 messages sont incorrectement classifiés dans *mésusage* et 9 dans *reste*. Parmi ces 12 messages, 4 contiennent des termes associés à l'excès et à des effets nocifs, comme dans *Je n'imaginais pas que c'était si grave* ou *s'il vous plait ne faites pas n'importe quoi*. Ils sont donc très similaires aux messages comportant des mésusages ;

— Pour l'expérience *3 catégories*, 14 messages sont incorrectement classifiés dans *pas d'usage*, 11 dans *usage normal* et 20 dans *mésusage*. Hormis le fait que ce modèle est plus complexe à résoudre, y compris pour les cas de mésusage, il ne permet pas de dégager d'explications sur les erreurs commises.

5 Conclusion

Ce travail propose un ensemble d'expériences pour détecter les mésusages de médicaments dans les messages de forums de discussion. Des forums francophones de *Doctissimo* sont exploités. Les messages sont d'abord prétraités et indexés avec un lexique adapté. Nous effectuons ensuite plusieurs séries d'expériences de catégorisation supervisée avec pour objectif de catégoriser chaque message dans une des trois catégories : *pas d'usage*, *usage normal* et *mésusage*. Les descripteurs exploités sont : le texte lemmatisé et vectorisé sans les mots grammaticaux, le code CIM-10 pour les maladies et le code ATC pour les médicaments, et leurs différentes variations et combinaisons.

Les meilleurs résultats sont obtenus avec l'expérience qui différencie entre les messages contenant un mésusage et le reste de messages. La F-mesure atteint alors jusqu'à 0,810 points. Les algorithmes bayésiens sont les plus efficaces face à cette tâche. Nous effectuons également une série d'expériences complémentaires pour déterminer l'impact des médicaments et des maladies présents dans le texte sur la qualité de la catégorisation. Il apparaît que les noms de médicaments et de maladies ont une influence positive sur les résultats, bien qu'elle soit faible.

La limitation principale de ce travail est que le nombre de messages décrivant des mésusages n'est pas élevé, ce qui réduit le potentiel de la catégorisation supervisée. Malgré ceci, la méthode proposée peut être utilisée pour détecter les messages contenant un mésusage de médicament. Dans l'avenir, nous utiliserons cette méthode pour enrichir la catégorie des mésusages et améliorer les modèles de catégorisation. Ensuite, une catégorisation plus fine pourra être effectuée pour différencier les types de mésusages, comme ceux proposés dans un travail existant (Bigeard *et al.*, 2018).

Remerciements

La présente publication s'inscrit dans le programme *Drugs Systematized Assessment in real-liFe Environnement (DRUGS-SAFE)* financé par l'Agence Nationale de Sécurité du Médicament et des Produits de Santé. Cette publication ne représente pas nécessairement l'opinion de l'ANSM.

Nous remercions Pierre Simonetti pour son aide avec l'annotation de messages et l'équipe ERIAS pour les conseils et le soutien.

Références

BIGEARD E. (2017). Construction de lexiques pour l'extraction de maladies dans les forums santé. In *RECITAL 2017*, p. 1–12.

BIGEARD E., GRABAR N. & THIESSARD F. (2018). Typology of drug misuse created from information available in health fora. In *MIE 2018*, p. 1–5.

BREIMAN L. (2001). Random forests. *Machine Learning*, **45**(1), 5–32.

CAMERON D., SMITH G. A., DANIULAITYTE R., SHETH A. P., DAVE D., CHEN L., ANAND G., CARLSON R., WATKINS K. Z. & FALCK R. (2013). PREDOSE : a semantic web platform for drug abuse epidemiology using social media. **46**(6), 985–997.

COHEN J. (1960). A coefficient of agreement for nominal scales. *Educational and Psychological Measurement*, **20**(1), 37–46.

COLLIER N. (2011). Towards cross-lingual alerting for bursty epidemic events. *J Biomed Semantics*, **2**(5), S10.

GAUDUCHEAU N. (2008). La communication des émotions dans les échanges médiatisés par ordinateur : bilan et perspectives. *Bulletin de psychologie*, p. 389–404.

JOHN G. H. & LANGLEY P. (1995). Estimating continuous distributions in bayesian classifiers. In M. KAUFMANN, Ed., *Eleventh Conference on Uncertainty in Artificial Intelligence*, p. 338–345, San Mateo.

KALYANAM J., KATSUKI T., LANCKRIET G. R. G. & MACKEY T. K. (2017). Exploring trends of nonmedical use of prescription drugs and polydrug abuse in the twittersphere using unsupervised machine learning. **65**, 289–295.

LANDIS J. & KOCH G. (1977). The measurement of observer agreement for categorical data. *Biometrics*, **33**, 159–174.

LANDWEHR N., HALL M. & FRANK E. (2005). Logistic model trees. *Machine Learning*, **95**(1-2), 161–205.

LEJEUNE G., BRIXTEL R., LECLUZE C., DOUCET A. & LUCAS N. (2013). Added-value of automatic multilingual text analysis for epidemic surveillance. In *Artificial Intelligence in Medicine (AIME)*.

LOUIS A. (2016). Natural language processing for social media. *Computational Linguistics*, **42**(4), 833–836.

MCCALLUM A. & NIGAM K. (1998). A comparison of event models for naive bayes text classification. In *AAAI workshop on Learning for Text Categorization*.

MORLANE-HONDÈRE F., GROUIN C. & ZWEIGENBAUM P. (2016). Identification of drug-related medical conditions in social media. In *LREC*, p. 1–7.

OMS (1995). *Classification statistique internationale des maladies et des problèmes de santé connexes — Dixième révision*. Organisation mondiale de la Santé, Genève.

QUINLAN J. (1993). *C4.5 Programs for Machine Learning*. San Mateo, CA : Morgan Kaufmann.

SCHMID H. (1994). Probabilistic part-of-speech tagging using decision trees. In *ICNMLP*, p. 44–49, Manchester, UK. treetagger.

SEBASTIANI F. (2002). Machine learning in automated text categorization. *ACM Computing Surveys*, **34**(1), 1–47.

SKRBO A., BEGOVIĆ B. & SKRBO S. (2004). Classification of drugs using the atc system (anatomic, therapeutic, chemical classification) and the latest changes. *Med Arh*, **58**(2), 138–41.

TAPI NZALI M. (2017). *Analyse des médias sociaux de santé pour évaluer la qualité de vie des patientes atteintes d'un cancer du sein*. Thèse de doctorat, Université de Montpellier, Montpellier, France.

WITTEN I. & FRANK E. (2005). *Data mining : Practical machine learning tools and techniques*. Morgan Kaufmann, San Francisco.

Utilisation de Représentations Distribuées de Relations pour la Désambiguïsation d'Entités Nommées

Nicolas Wagner Romaric Besançon Olivier Ferret
CEA, LIST, Laboratoire Vision et Ingénierie des Contenus, Gif-sur-Yvette, F-91191, France.
{romaric.besancon,olivier.ferret}@cea.fr

RÉSUMÉ

L'identification des entités nommées dans un texte est une étape fondamentale pour de nombreuses tâches d'extraction d'information. Pour avoir une identification complète, une étape de désambiguïsation des entités similaires doit être réalisée. Celle-ci s'appuie souvent sur la seule description textuelle des entités. Or, les bases de connaissances contiennent des informations plus riches, sous la forme de relations entre les entités : cette information peut également être exploitée pour améliorer la désambiguïsation des entités. Nous proposons dans cet article une approche d'apprentissage de représentations distribuées de ces relations et leur utilisation pour la tâche de désambiguïsation d'entités nommées. Nous montrons le gain de cette méthode sur un corpus d'évaluation standard, en anglais, issu de la tâche de désambiguïsation d'entités de la campagne TAC-KBP.

ABSTRACT

Exploiting Relation Embeddings to Improve Entity Linking

The identification of named entities in documents is the basis of most Information Extraction tasks. The full identification of a named entity includes its disambiguation, which generally relies on its textual description only. However, the relations between entities in knowledge bases are also an interesting source of information that can be exploited for performing such disambiguation. In this article, we propose to build word embeddings for representing such relations and to use these representations for disambiguating entity mentions in texts. Furthermore, we show the interest of this approach on the 2015 TAC-KBP corpus, a standard corpus for evaluating Entity Linking systems.

MOTS-CLÉS : Désambiguïsation d'entités nommées, apprentissage de représentations, relations.

KEYWORDS: Entity linking, named entities, embeddings, relations.

1 Introduction

Dans le domaine de l'extraction d'information à partir de textes, la reconnaissance d'entités nommées prend une place particulièrement importante, d'une part parce qu'elle peut servir à représenter le contenu thématique de certains documents de façon informative et concise (de qui on parle, où se situe l'action ...) et d'autre part parce que ces entités sont en général nécessaires à tous les traitements plus complexes qui peuvent être entrepris à la suite, comme l'extraction de relations ou d'événements, ou le remplissage automatique de bases de connaissances.

De ce fait, la qualité et la précision de la reconnaissance des entités nommées est fondamentale. Une étape additionnelle augmentant la précision de l'identification des entités consiste à lever l'ambiguïté

entre des entités ayant des mentions identiques ou similaires. Cette étape de désambiguïsation d'entités nommées (aussi appelée liaison référentielle d'entités ou *Entity Linking*) s'appuie sur l'utilisation d'une base de connaissances contenant les entités connues *a priori* servant de référence pour la désambiguïsation (Shen *et al.*, 2015; Ling *et al.*, 2015). Par exemple, la mention d'entité « *Castro* » correspond à deux entités différentes dans les contextes suivants [1] :

*Obama and **Castro** shake hands as U.S. and Cuba seek better ties.* → Raúl Castro
*Clinton/**Castro** would blow up Jeb's dream of aligning with the Hispanic voters.* → Julián Castro

Traditionnellement, cette désambiguïsation s'appuie sur la similarité entre le contexte textuel de la mention de l'entité à désambiguïser et une description textuelle associée aux entités dans la base de connaissances (le plus souvent, le texte de la page Wikipédia associée à l'entité). Or, les bases de connaissances possèdent également une structure relationnelle associant les entités entre elles. Plusieurs études ont montré que ces relations peuvent être exploitées pour aider la désambiguïsation, que ce soit dans un cadre général de désambiguïsation lexicale (Moro *et al.*, 2014) ou dans le cadre plus spécifique de la désambiguïsation des entités nommées (Usbeck *et al.*, 2014), en particulier pour celles dont le contenu textuel associé est pauvre ou inexistant (Besançon *et al.*, 2016).

Par ailleurs, des travaux récents ont montré que l'usage de représentations sémantiques distribuées (*word embeddings*) sur le modèle de *word2vec* (Mikolov *et al.*, 2013), construites pour représenter de façon jointe les mots des textes et les entités nommées d'une base de connaissances, peut améliorer la détection de la similarité entre les mentions d'entités et les entités de cette base, augmentant ainsi la précision de la désambiguïsation (Fang *et al.*, 2016; Yamada *et al.*, 2016; Moreno *et al.*, 2017).

Nous proposons, dans cette étude, d'associer ces deux idées en apprenant des représentations distribuées des relations entre les entités d'une base de connaissances et en exploitant ces représentations pour aider la désambiguïsation des entités nommées.

2 Approche

2.1 Apprentissage de représentations des relations entre entités

La désambiguïsation des entités nommées se fait en utilisant une base de connaissances contenant des entités auxquelles sont associées un ensemble d'informations. Parmi les bases de connaissances traditionnellement utilisées pour cette tâche, on peut citer YAGO (Suchanek *et al.*, 2007), DBPedia (Lehmann *et al.*, 2015), Freebase (Bollacker *et al.*, 2008) ou Wikidata (Vrandečić & Krötzsch, 2014). Les informations associées aux entités prennent la forme de propriétés ou de relations typées entre les entités : par exemple, à une entité de type personne pourra se voir associer son *âge* comme propriété et son *lieu de naissance* comme relation avec une autre entité, de type Lieu, également présente dans la base. Certaines relations peuvent également impliquer plus de deux entités (par exemple, une relation de type *mariage* associe deux personnes, un lieu et une date).

Notre objectif est d'exploiter toutes ces relations entre entités (binaires ou n-aires) existant dans la base de connaissances de référence pour aider la désambiguïsation des entités. Ces relations fournissent en effet une information importante concernant les entités. Pour reprendre l'exemple précédent sur la

1. Les exemples de textes, d'entités et de relations présentés dans l'article sont extraits des données de la campagne d'évaluation TAC-KBP pour la tâche de désambiguïsation d'entités nommées (EDL).

mention « *Castro* », les relations pour les deux entités « *Raúl Castro* » et « *Julián Castro* », présentées dans le tableau 1, montrent des éléments pouvant aider la désambiguïsation : par exemple, le lien avec Cuba pour le premier et celui avec le parti démocrate américain pour le second.

Relation	**Raúl Castro**	**Julián Castro**
party	Communist Party of Cuba	Democratic Party
position	Prime Minister of Cuba	Mayor of San Antonio
jurisdiction	Cuba	San Antonio
spouse	vilma espín	Erica Lira Castro
education	Belen Jesuit Preparatory School	harvard law school
sibling	Fidel Castro	Joaquín Castro

TABLE 1 – Exemple de relations pour les entités « *Raúl Castro* » et « *Julián Castro* »

Dans ce but, nous proposons de définir une similarité relationnelle entre une mention d'entité et les entités de la base de connaissances en nous appuyant sur l'apprentissage de représentations jointes entre les mots et les entités, fondées sur leur contexte relationnel, afin de déterminer si une mention et une entité candidate sont compatibles du point de vue de leurs relations.

Plusieurs modèles pour l'apprentissage de représentations de relations (*embeddings* de relations) ont été proposés (Nickel *et al.*, 2012; Riedel *et al.*, 2013; Weston *et al.*, 2013), mais principalement dans le but d'améliorer l'extraction de relations : leur objectif est donc d'estimer la probabilité de l'existence d'une relation entre deux entités dans une base de connaissances. Nous nous plaçons pour notre part dans une optique différente, où nous souhaitons comparer les contextes relationnels de deux éléments. De plus, nous voulons avoir la possibilité d'apprendre, de façon jointe, des représentations pour les mots et les entités en fonction de leur relations. Nous proposons donc d'utiliser un apprentissage de représentations s'appuyant sur *word2vec*, mais en utilisant des contextes relationnels au lieu de contextes textuels, de manière similaire au modèle DeepWalk (Perozzi *et al.*, 2014).

Dans (Moreno *et al.*, 2017), un apprentissage joint de représentations de mots et d'entités est déjà réalisé en exploitant les contextes textuels des entités. Nous proposons ici d'étendre cette idée aux contextes relationnels : à partir du graphe des relations impliquant une entité, ces contextes sont construits comme des séquences d'entités en relation obtenues selon un modèle de marche aléatoire dans ce graphe comparable à celui de DeepWalk. Dans notre cas néanmoins, nous initialisons deux marches aléatoires à partir d'une entité pour construire un contexte gauche et un contexte droit de l'entité, ce qui permet d'avoir un nombre fixe de contextes pour chaque entité.

Plus précisément, soit e une entité de la base de connaissances, associée à un identifiant unique $id(e)$ et un nom $nom(e)$, non nécessairement unique puisque correspondant à la chaîne de caractères d'une mention de e. Soit également $Rel(e) = \{e_1, e_2, e_3, ..., e_n\}$, l'ensemble des entités en relation avec e. On construit un ensemble de séquences ayant e comme élément central en sélectionnant aléatoirement deux entités e_i et $e_j \in Rel(e)$ et en les plaçant comme contexte immédiat gauche et droit de e, puis en itérant à partir de e_i et e_j. Au deuxième niveau, on obtient donc une séquence du type :

$$e_k \in Rel(e_i) \leftrightarrow e_i \in Rel(e) \leftrightarrow \boldsymbol{e} \leftrightarrow e_j \in Rel(e) \leftrightarrow e_l \in Rel(e_j)$$

À partir de cette séquence, deux contextes sont construits, un utilisant l'identifiant de l'entité, l'autre utilisant le nom de l'entité :

$$nom(e_k)\ nom(e_i)\ \boldsymbol{nom(e)}\ nom(e_j)\ nom(e_l)$$
$$nom(e_k)\ nom(e_i)\ \boldsymbol{id(e)}\ nom(e_j)\ nom(e_l)$$

En reprenant l'exemple du tableau 1 et pour une taille de contexte de 1, on générerait pour l'entité « *Raúl Castro* » (associée à l'identifiant *m.01gcjq*) donc des contextes du type :

Communist Party of Cuba	*Raúl Castro*	*Cuba*
Fidel Castro	*Raúl Castro*	*Prime Minister of Cuba*
Communist Party of Cuba	*m.01gcjq*	*Cuba*
Fidel Castro	*m.01gcjq*	*Prime Minister of Cuba*

...

Un modèle *word2vec* est alors entraîné à partir de ces contextes. Le fait de générer deux pseudo-phrases pour chacune des entités, une avec son nom et l'autre avec son identifiant, permet de créer une représentation jointe d'entités et de mots.

2.2 Similarité entre mention et entité

L'apprentissage de représentation pour les mentions et les entités décrit à la section précédente nous donne la possibilité de comparer une mention et une entité directement par une mesure de similarité entre leurs représentations. Néanmoins, cette comparaison peut être enrichie en prenant en compte le contexte relationnel de la mention d'entité. Ce dernier, appelé dans ce qui suit contexte relationnel textuel, est défini par l'ensemble des mentions d'entités proches de la mention cible dans le texte. Pour la phrase ci-dessus « *Obama and **Castro** shake hands as U.S. and Cuba seek better ties.* » par exemple, le contexte relationnel textuel de *Castro* est ainsi formé par les mentions {*Obama, U.S., Cuba*}. Cette notion constitue le pendant textuel du contexte relationnel d'une entité au sein de la base de connaissances. À la différence de celui-ci cependant, le contexte relationnel textuel n'est pas fondé sur des relations typées explicites mais repose sur une simple relation de cooccurrence textuelle entre la mention cible et les mentions d'entités qui l'environnent. Nous faisons l'hypothèse que cette cooccurrence est la manifestation de relations entre les entités auxquelles réfèrent les mentions considérées.

L'exploitation conjointe des représentations des mentions, des entités et de la notion de contexte relationnel textuel dans le cadre défini par (Moreno *et al.*, 2017) nous a conduit à définir les quatre mesures de similarité suivantes entre une mention et une entité candidate :

- REL_1 la similarité directe entre le vecteur de la mention et de celui de l'entité candidate, mesurée par une distance cosinus ;
- REL_2 la similarité cosinus entre le vecteur moyen des mentions du contexte relationnel textuel et le vecteur de l'entité candidate ;
- REL_3 la valeur moyenne des similarités cosinus entre le vecteur de chaque mention du contexte relationnel textuel et le vecteur de l'entité candidate ;
- REL_4 la moyenne des k meilleures similarités parmi celles considérées pour REL_3.

En notant $\vec{m}$ le vecteur de la mention, $\vec{e}$ le vecteur de l'entité et $p(m)$ le contexte relationnel de la mention, ces mesures se définissent de façon plus formelle par les équations suivantes :

$$REL_1(e,m) = cos(\vec{e},\vec{m}) \qquad REL_2(e,m) = cos(\vec{e}, \frac{\Sigma_{w_i \in p(m)} \vec{w_i}}{||p(m)||})$$

$$REL_3(e,m) = \frac{\Sigma_{w_i \in p(m)} cos(\vec{e},\vec{w_i})}{||p(m)||} \qquad REL_4(e,m) = \frac{\Sigma argmax_{w_i \in p(m) | i=1,\ldots,k} cos(\vec{e},\vec{w_i})}{k}$$

2.3 Intégration dans un système général de désambiguïsation d'entités

Pour tester cette approche, nous intégrons le score généré par ces représentations dans un système global de désambiguïsation d'entités s'appuyant sur un apprentissage supervisé pour l'étape de désambiguïsation proprement dite. Ce système suit une architecture standard en deux grandes étapes :

— une étape de génération des entités candidates : à partir d'une mention d'entité, un ensemble d'entités candidates de la base de connaissances est généré en s'appuyant sur l'égalité ou la proximité de la mention avec le nom de l'entité ou un nom (Dredze *et al.*, 2010) ;

— une étape de sélection de la meilleure entité candidate, fondée sur un apprentissage supervisé. Cet apprentissage utilise les couples (mention, entité candidate) comme exemples positifs lorsque l'entité candidate est l'entité de référence et comme exemples négatifs sinon. Les exemples sont représentés par un vecteur de traits comprenant des traits indiquant la façon dont le candidat a été généré (*i.e.* le degré de proximité entre les chaînes de caractères de la mention et de l'entité candidate) ainsi que plusieurs mesures de similarité : une similarité cosinus entre les représentations vectorielles du document et de la description textuelle de l'entité et les similarités fondées sur les représentations jointes de mots et d'entités proposées par (Moreno *et al.*, 2017).

Ce système forme notre modèle de base (noté *baseline* dans les résultats). Le score de similarité relationnelle y est intégré par l'ajout d'un ou plusieurs traits supplémentaires.

3 Évaluation

3.1 Corpus et mesures

Pour l'évaluation de nos méthodes, nous utilisons les données de la tâche de désambiguïsation d'entités nommées (EDL, pour *Entity Discovery and Linking*) de la campagne TAC KBP de 2015 [2]. Nous présentons dans le tableau 2 des statistiques sur ces données. La base de connaissances de référence utilisée pour lever l'ambiguïté des entités est une sous-partie de la base Freebase (Bollacker *et al.*, 2008), contenant plus de 8 millions d'entités. Chaque document des collections d'entraînement (*train*) et de test contient un ensemble de mentions d'entités annotées avec les entités auxquelles elles font référence dans Freebase. Lorsqu'aucune entité de la base de connaissances ne correspond à la mention, celle-ci est annotée en tant que *NIL*.

TAC 2015	Nb. docs	Nb. mentions	Nb. mentions NIL	Nb. candidats	Nb candidats NIL
train	168	12 175	3 215	1 722 518	1 745
test	167	13 587	3 379	1 720 767	1 751

TABLE 2 – Description des données de la campagne TAC KBP 2015

2. https://tac.nist.gov//2015/KBP/

Pour chacune des mentions d'entraînement et de test, la génération des candidats telle que décrite à la section 2.3 permet d'obtenir un certain nombre d'entités candidates. Si pour une mention donnée, il n'a été trouvé aucun candidat potentiel, elle est classée comme *NIL*. Nous rapportons également le nombre de candidats et de NIL générés dans le tableau 2.

Pour évaluer les performances de notre modèle, nous utilisons les mesures de précision (P) et rappel (R) sur le candidat sélectionné par le système, s'il existe (link), ou sur les réponses NIL sinon. Ces mesures sont combinées de façon standard par un f-score (F). Si l'on note e_r, l'entité de référence associée à la mention m, e_t l'entité associée à m par notre système et $N(x)$, le nombre de mentions d'entités qui vérifiant l'expression x, les mesures utilisées se définissent par :

$$P(NIL) = \frac{N(e_t{=}NIL \ \&\ e_r{=}NIL)}{N(e_t{=}NIL)} \qquad P(link) = \frac{N(e_t{=}e_r \ \&\ e_t{\neq}NIL)}{N(e_t{\neq}NIL)} \qquad P(all) = \frac{N(e_t{=}e_r)}{N(e_t)}$$
$$R(NIL) = \frac{N(e_t{=}NIL \ \&\ e_r{=}NIL)}{N(e_r{=}NIL)} \qquad R(link) = \frac{N(e_t{=}e_r \ \&\ e_t{\neq}NIL)}{N(e_r{\neq}NIL)}$$

Notons que $P(all) = R(all) = F(all)$ si le classifieur a donné une réponse pour toutes les mentions ($N(e_t) = N(e_r)$).

Le système de classification supervisé utilisé pour la désambiguïsation s'appuie sur un modèle non déterministe (en particulier pour faire des échantillonnages aléatoires des données d'entraînement). Les scores présentés dans cette étude sont des moyennes des scores obtenus sur 10 tests. Pour avoir une comparaison correcte des différents modèles, chacun des tests est réalisé avec la même graine aléatoire pour les différents modèles.

3.2 Résultats

Les résultats présentés dans cette section ont été obtenus avec une taille du contexte relationnel des entités limité à 1 : on ne considère donc que des séquences de trois éléments pour la construction des représentations avec *word2vec*. Cette construction a été réalisée avec le modèle CBOW, une taille des représentations égale à 400 et une seule itération. Au niveau du système global de désambiguïsation, le classifieur utilisé pour la sélection de la meilleure entité candidate est Adaboost.

L'intégration dans le système des 4 similarités relationnelles (REL_1, REL_2, REL_3 et REL_4) a été testée de façon indépendante (ajout d'une des similarités comme trait pour le classifieur) ou combinée (ajout de plusieurs de ces similarités comme traits pour le classifieur). Nos expériences ont en effet montré que combiner les mesures peut apporter des améliorations.

	P(all)	P(NIL)	R(NIL)	F(NIL)	P(link)	R(link)	F(link)
baseline	0,765	0,624	0,908	0,74	0,845	0,718	0,776
REL_1	0,776	**0,626**	0,909	0,741	0,861	0,732	0,792
$REL_{1,2}$	**0,781**	**0,626**	**0,92**	**0,745**	**0,871**	**0,735**	**0,797**
$REL_{1,2,3,4}$	0,773	0,614	0,915	0,735	0,867	0,727	0,79
$REL_{1,2}$ + type d'entité	**0,742**	–	–	–	–	–	–
Top TAC-EDL 2015	0,737	–	–	–	–	–	–
(Moreno *et al.*, 2017)	**0,742**	–	–	–	–	–	–

TABLE 3 – Résultats de désambiguïsation d'entités sur TAC-EDL 2015, sur une moyenne de 10 runs

Nous présentons dans le tableau 3 les résultats les plus significatifs. Dans tous les cas, nous observons que l'ajout d'une mesure de similarité utilisant les représentations des relations de la base de connaissances permet d'améliorer les résultats par rapport au modèle de base. Parmi les différentes mesures de similarité considérées de façon indépendante, la mesure directe REL_1 est la plus performante, avec une amélioration du score global de 1,1 points. Combiner les mesures de similarité permet d'améliorer encore les résultats mais conjuguer de façon naïve la totalité des mesures donne des résultats inférieurs à ceux obtenus avec la seule mesure REL_1. La meilleure configuration associe en fait les mesures REL_1 et REL_2, avec un gain de 1,6 points par rapport à la *baseline*. On peut ainsi supposer que combiner une mesure directe et une mesure prenant en compte un contexte un peu plus large de la mention est intéressant à condition toutefois que l'influence de ce contexte ne noie pas trop l'information apportée par la mesure directe.

Dans le détail, on constate que le gain porte surtout sur les scores de type *link* (précision et rappel), c'est-à-dire que le modèle aide à repérer les bonnes entités lorsqu'elles existent. Mais on observe également que l'amélioration sur le rappel pour les entités NIL augmente de façon notable. De façon générale, la différence sur les entités NIL entre le score élevé pour le rappel et un moins bon score pour la précision montre que l'approche définie tend à classer un peu trop d'entités comme NIL.

Nous comparons également nos résultats à l'état de l'art. Les mesures officielles de la campagne se fondent à la fois sur la détection de l'entité et sur la détection du type de l'entité (y compris pour les entités NIL). Les scores rapportés ci-dessus n'incluent pas cette notion de type. En ajoutant cette contrainte, notre meilleur modèle obtient un score de 0,742. Le meilleur participant à la campagne TAC EDL 2015 avait un score global de 0,737 et Moreno *et al.* (2017) rapportent également, sur cette collection, un score de 0,742. Les résultats obtenus ici sont donc comparables à l'état de l'art. On note que Moreno *et al.* (2017) utilisent également d'autres indices, telles que la popularité *a priori* d'une entité, qui ne sont pas exploités ici et que l'on pourrait ajouter à notre système comme traits supplémentaires.

4 Conclusion et perspectives

Nous présentons dans cet article une approche pour l'exploitation des relations entre les entités présentes dans une base de connaissances pour aider la désambiguïsation des entités nommées. Cette méthode s'appuie sur un apprentissage de représentations pour les relations (*relation embeddings*) exploitant le modèle *word2vec* sur les contextes relationnels des entités dans la base de connaissances. En évaluant ce modèle sur les données de la campagne d'évaluation TAC EDL 2015, nous montrons que l'ajout de cette connaissance permet d'améliorer les performances d'un système de désambiguïsation d'entités de façon significative. Comme nous l'avons montré, la prise en compte d'un certain contexte autour d'une mention d'entité est intéressante mais doit être contrôlée soigneusement. Une extension possible de ce travail est de considérer le contexte à un niveau plus large, en l'occurrence celui du document, en réalisant de façon jointe la désambiguïsation d'un ensemble de mentions à l'instar des travaux menés à la suite de (Cucerzan, 2007) ou de (Kulkarni *et al.*, 2009) par exemple.

Références

BESANÇON R., DAHER H., FERRET O. & BORGNE H. L. (2016). Utilisation des relations d'une

base de connaissances pour la désambiguïsation d'entités nommées. In *Actes de la 23e conférence sur le Traitement Automatique des Langues Naturelles*, p. 290–303, Paris, France: Association pour le Traitement Automatique des Langues.

BOLLACKER K., EVANS C., PARITOSH P., STURGE T. & TAYLOR J. (2008). Freebase: A Collaboratively Created Graph Database for Structuring Human Knowledge. In *Proceedings of the 2008 ACM SIGMOD International Conference on Management of Data*, p. 1247–1250, Vancouver, Canada: ACM.

CUCERZAN S. (2007). Large-scale named entity disambiguation based on Wikipedia data. In *Proceedings of the issue2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL 2007)*, p. 708–716, Prague, Czech Republic.

DREDZE M., MCNAMEE P., RAO D., GERBER A. & FININ T. (2010). Entity Disambiguation for Knowledge Base Population. In *Proceedings of the 23rd International Conference on Computational Linguistics (COLING'10)*, p. 277–285, Beijing, China: Association for Computational Linguistics.

FANG W., ZHANG J., WANG D., CHEN Z. & LI M. (2016). Entity Disambiguation by Knowledge and Text Jointly Embedding. In *Proceedings of the 20th SIGNLL Conference on Computational Natural Language Learning (CoNLL 2016)*, p. 260–269, Berlin, Germany.

KULKARNI S., SINGH A., RAMAKRISHNAN G. & CHAKRABARTI S. (2009). Collective Annotation of Wikipedia Entities in Web Text. In *Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD 2009)*, p. 457–466, Paris, France: ACM.

LEHMANN J., ISELE R., JAKOB M., JENTZSCH A., KONTOKOSTAS D., MENDES P., HELLMANN S., MORSEY M., VAN KLEEF P., AUER S. & BIZER C. (2015). DBpedia – A large-scale, multilingual knowledge base extracted from wikipedia. *Semantic Web Journal*, **6**(2), 167–195.

LING X., SINGH S. & WELD D. (2015). Design Challenges for Entity Linking. *Transactions of the Association for Computational Linguistics (TACL)*, **3**, 315–328.

MIKOLOV T., SUTSKEVER I., CHEN K., CORRADO G. S. & DEAN J. (2013). Distributed Representations of Words and Phrases and their Compositionality. In *Advances in Neural Information Processing Systems 26*, p. 3111–3119. Curran Associates, Inc.

MORENO J., BESANÇON R., BEAUMONT R., D'HONDT E., LIGOZAT A.-L., ROSSET S., TANNIER X. & GRAU B. (2017). Combining Word and Entity Embeddings for Entity Linking. In *The Semantic Web. ESWC 2017*, volume 10249 of *Lecture Notes in Computer Science*: Springer.

MORO A., RAGANATO A. & NAVIGLI R. (2014). Entity Linking meets Word Sense Disambiguation: a Unified Approach. *Transactions of the Association for Computational Linguistics (TACL)*, **2**, 231–244.

NICKEL M., TRESP V. & KRIEGEL H.-P. (2012). Factorizing YAGO: scalable machine learning for linked data. In *Proceedings of the 21st international conference on World Wide Web (WWW 2012)*, p. 271–280, Lyon, France: ACM.

PEROZZI B., AL-RFOU R. & SKIENA S. (2014). DeepWalk: Online Learning of Social Representations. In *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD 2014)*, p. 701–710, New York, USA: ACM.

RIEDEL S., YAO L., MCCALLUM A. & MARLIN B. M. (2013). Relation Extraction with Matrix Factorization and Universal Schemas. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL HLT 2013)*, p. 74–84, Atlanta, Georgia, USA.

SHEN W., WANG J. & HAN J. (2015). Entity Linking with a Knowledge Base: Issues, Techniques, and Solutions. *Transactions on Knowledge & Data Engineering*, **27**(2), 443–460.

SUCHANEK F. M., KASNECI G. & WEIKUM G. (2007). Yago: A Core of Semantic Knowledge. In *Proceedings of the 16th International Conference on World Wide Web (WWW 2007)*, p. 697–706, Banff, Alberta, Canada: ACM.

USBECK R., NGOMO A.-C. N., RÖDER M., GERBER D., COELHO S., AUER S. & BOTH A. (2014). AGDISTIS - Graph-Based Disambiguation of Named Entities using Linked Data. In *The Semantic Web – ISWC 2014*, volume 8796 of *Lecture Notes in Computer Science*, p. 457–471. Springer International Publishing.

VRANDEČIĆ D. & KRÖTZSCH M. (2014). Wikidata: A Free Collaborative Knowledgebase. *Communications of the ACM*, **57**(10), 78–85.

WESTON J., BORDES A., YAKHNENKO O. & USUNIER N. (2013). Connecting Language and Knowledge Bases with Embedding Models for Relation Extraction. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing (EMNLP 2013)*, p. 1366–1371, Seattle, Washington, USA.

YAMADA I., SHINDO H., TAKEDA H. & TAKEFUJI Y. (2016). Joint Learning of the Embedding of Words and Entities for Named Entity Disambiguation. In *Proceedings of the 20th SIGNLL Conference on Computational Natural Language Learning (CoNLL 2016)*, p. 250–259, Berlin, Germany.

Traduction automatique du japonais vers le français
Bilan et perspectives

Raoul Blin
CNRS-CRLAO, 105 Raspail, 31000 Paris, France
blin@ehess.fr

RÉSUMÉ
Nous étudions la possibilité de construire un dispositif de traduction automatique neuronale du japonais vers le français, capable d'obtenir des résultats à la hauteur de l'état de l'art, sachant que l'on ne peut disposer de grands corpus alignés bilingues. Nous proposons un état de l'art et relevons de nombreux signes d'amélioration de la qualité des traductions, en comparaison aux traductions statistiques jusque-là prédominantes. Nous testons ensuite un des baselines librement disponibles, OpenNMT, qui produit des résultats encourageants. Sur la base de cette expérience, nous proposons plusieurs pistes pour améliorer à terme la traduction et pour compenser le manque de corpus.

ABSTRACT
Machine Translation from Japanese to French - Review and Prospects.

In this paper, we discuss the feasability of a neural machine translation system from Japanese to French, which would be able to obtain results that are equivalent to the state of the art. We first describe the state of the art for Ja>Fr and see many improvements compared to statistical machine translation. Second, we evaluate one free open baseline, OpenNMT, which achieves encouraging results. Based on these experiments, we suggest some ways to improve the results, taking into account the lack of improved and large corpora.

MOTS-CLÉS : traduction automatique ; japonais ; français.

KEYWORDS: machine translation ; Japanese ; French.

On peut affirmer que jusqu'en 2015 à peu près, les rares dispositifs de traduction automatique (TA) du japonais vers le français, à base de statistiques (TAS), produisaient de piètres résultats. Les traductions permettaient tout au plus de se faire une idée du thème et des relations logiques entre les entités (individus, concepts, événements etc.) correctement identifiées, pour des textes relativement simples. Les très rares évaluations chiffrées disponibles confortent ce sentiment.

Depuis, en très peu de temps, de nombreux bouleversements ont eu lieu dans le domaine de la traduction automatique en général, qui peuvent avoir une incidence sur la traduction du ja>fr en particulier : introduction de la traduction automatique neuronale (TAN), multiplication des dispositifs librement accessibles, dont des dispositifs «clef en main» et libres performants, traitement multilingue capable de compenser le manque de corpus. Les cartes sont donc rebattues aussi pour le ja>fr. Nous nous interrogeons dans le présent article sur la possibilité de construire un dispositif de traduction du japonais vers le français capable de produire des traductions de niveau «veille» [1], sachant que l'on ne

1. Nous reprenons la distinction entre qualité de traduction de niveau «veille» et de niveau «fluide». La traduction de «veille» permet de comprendre le thème et les relations logiques entre entités, événements, concepts. Il n'y a pas d'exigence sur la qualité de la formulation (registre de langue, grammaticalité). La lecture peut donc être pénible. Une traduction «fluide»

peut disposer de corpus ja>fr de très grande taille.

Nous procédons en deux temps. Tout d'abord, nous dressons un état de l'art pour le couple ja>fr. Dans la deuxième partie, nous testons un baseline de TAN élaboré à partir d'outils et ressources disponibles et libres. Cela nous permet d'estimer la qualité accessible pour un dispositif «autonome» (non dépendant de systèmes commerciaux ou autres). Nous proposons pour finir un bilan et les perspectives d'amélioration.

Nos évaluations sont essentiellement basées sur le score BLEU (Papineni *et al.*, 2002) et sont de ce fait assez grossières. Elles suffisent néanmoins à dessiner une tendance générale. Nous sommes amenés à évaluer les productions des services commerciaux en ligne. Nous gardons à l'esprit que ces évaluations sont biaisées car il est impossible de connaître la taille et le contenu des corpus utilisés pour l'entraînement, ainsi que les pré- et post-traitement appliqués. Comme il existe peu d'évaluations chiffrées pour les systèmes de TA ja>fr, nous sommes obligé de comparer les résultats avec ceux des traductions vers l'anglais. Là aussi, nous gardons à l'esprit que le score BLEU est minoré pour le français. Ainsi, si une erreur concerne le nombre du groupe nominal, en français cette erreur sera répercutée sur les déterminants, adjectifs et autres structures qui dépendent de ce nom. Par contre, elle ne sera comptée qu'une fois pour l'anglais, qui sera donc moins pénalisé.

1 Ressources utilisées pour les tests

Les évaluations présentées dans les deux sections suivantes portent sur plusieurs corpus. Pour ce qui est de la thématique, aucun n'est spécialisé, sauf l'article isolé du Asahi. KFFT («WikiKyoto») (Neubig *et al.*, 2012) est un corpus japonais-anglais. Il regroupe des pages de Wikipédia sur Kyoto. Le texte est segmenté à l'aide de KyTea [2] (Neubig *et al.*, 2011), avec un entraînement optimisé pour ce corpus. L'unité d'alignement est la phrase. Le corpus OPUS-jafr est constitué du sous-ensemble des textes japonais-français bien encodés du corpus OPUS (Tiedemann & Nygaard, 2004). C'est le plus volumineux corpus bilingue aligné ja-fr, librement accessible. Les unités d'alignement sont variées. Il contient de nombreuses erreurs d'alignement (Blin, 2018).

ALIGNJaFr [3] reprend quelques sous-corpus de OPUS-jafr parmi ceux bien alignés et dont la traduction est la plus naturelle. D'autres textes y ont été ajoutés, ainsi qu'un lexique de mots et de locutions. Ces deux derniers n'ont cependant pas été utilisés pour constituer les corpus de validation et de tests. Une partie des textes ajoutés sont extraits du corpus «jibiki.com» et ont été peu ou prou réalignés manuellement. L'alignement exploite différentes unités : mots, syntagmes, phrases. La version 0.5 comprend des sous-corpus de validation et de test constitués de segments extraits de chaque sous-corpus selon un nombre proportionnel à la taille de chaque sous-corpus. AlignJaFrtitres diffère de la version 0.5 sur le corpus de test, composé des titres de presse utilisés par (Blin, 2014). Quelques titres ont été rajoutés. Les titres proviennent à parts à peu près égales, des cinq journaux nationaux généralistes à plus grand tirage japonais : Asahi, Mainichi, Nikkei, Sankei, Yomiuri. Pour chaque titre deux traductions sont proposées. La première est une traduction «de veille». Elle suit au mieux la structure du titre japonais et de ce fait est parfois peu naturelle en français. La seconde traduction est une traduction «fluide», dont la structure peut s'éloigner sensiblement de la structure originale japonaise. Les autres titres de presse présents dans le corpus sont traduits d'une seule manière ou

assure un niveau de veille et la bonne qualité de la formulation.

2. http ://www.phontron.com/kytea
3. https ://sharedocs.huma-num.fr/wl/ ?id=DDF9YZrqn6gWA6sqLTPPsoDs2YBmIRgk

non. Il semblerait que d'autres corpus alignés existent comme par exemple le BTEC (Takezawa *et al.*, 2002). Mais ceux-ci sont pour des raisons variés, inaccessibles et donc inutilisables.

Nous proposons en plus l'analyse qualitative d'un bref article de presse [4] édité en ligne par le journal Asahi. Il a été choisi au hasard (dernier article paru lorsque nous avons consulté la liste des titres). Il s'agit d'un texte relativement simple, évoquant un soupçon de fraude à la loi électorale. Il contient quelques termes techniques relatifs au droit et à l'administration.

Corpus	lgue(s)	Train	Dev	Test	
WikiKyoto	ja > an	330k phrases	1 235 ph.	1 160 ph.	19 mots/ph.
AlignJaFr 0.5	ja - fr	91k tokens	1 993 tok.	1 973 tok.	9 mots/tok.
AlignJaFrtitres 0.1	ja - fr	73k tok.	1 817 tok.	358/383 titres	16,5 mots/titre
OPUS-jafr	ja - fr	547K tok.	500 tok.	5 000	
Article Asahi	ja			10 phrases	24 mots/ph.
PresseMarchal	ja	379K titres			

TABLE 1 – Corpus utilisés.

2 Etat de l'art

L'ensemble de la communauté a salué les progrès opérés par la traduction automatique durant les toutes dernières années grâce à la TAN. Si les chiffres le confirment indéniablement pour les paires de langues comme les langues européennes (voir entre autres (Wu *et al.*, 2016)), la démonstration reste à faire pour la traduction du japonais vers le français, deux langues qui constituent une paire faiblement dotée en corpus bilingues alignés, de grande taille et de bonne qualité.

Evaluer l'évolution avant et après l'émergence de la TAN est difficile. A notre connaissance il n'existe que trois évaluations chiffrées disponibles pour dépeindre l'état de l'art pour le couple ja>fr du milieu des années 2010, à l'apogée du paradigme statistique, juste avant que la TAN ne s'impose comme le nouveau paradigme dominant. Malheureusement, la particularité des corpus utilisés pour faire ces évaluations réduisent la représentativité des résultats, mais il faut s'en contenter, faute d'alternatives. Les deux premières évaluations (sous la direction de Y.Lepage, Université de Waseda, Japon ; non publiées) ont porté sur le corpus OPUS-jafr. L'objectif était moins d'évaluer les outils de traduction que d'estimer, indirectement, la qualité des corpus. Les auteurs ont pour cela évalué une première traduction faite avec le service Google Translate, alors considéré comme l'état de l'art (Tableau 2-1). Une deuxième évaluation portait sur le baseline classique statistique chaîne à chaîne (GIZA++, Moses), le plus utilisé à l'époque dans le monde académique (Tableau 2-2). La contre-performance du traducteur de Google pouvait s'expliquer par la mauvaise qualité du corpus OPUS-jafr, dont était extrait le corpus de test (nombreux décalages dans les alignements ; faiblesse de certaines traductions). La troisième évaluation (Blin, 2014) portait sur la traduction des titres de presse par Google Translate et Bing Translator (Tableau 2-3). Peu de traductions de titres atteignaient le niveau veille. Deux explications sont possibles. Les titres de presses sont sous représentés dans les corpus d'entrainement. Google Translate traduisait du japonais vers le français via l'anglais, ce qui avait pour conséquence de démultiplier les erreurs.

4. https ://www.asahi.com/articles/ASL1Y6HPDL1YUTIL05V.html

Ces rares chiffres sont à comparer avec ceux obtenus pour d'autres paires de langues incluant le japonais. La paire ja-an était, et reste, à la fois la mieux dotée en corpus, la plus étudiée et la mieux documentée. Elle nous sert donc de référence. La comparaison est d'autant plus intéressante qu'anglais et français partagent plusieurs propriétés linguistiques qui les distinguent de la même manière du japonais : l'ordre des mots dans la phrase (SVO contre SOV japonais), marques casuelles préposées contre marques postposées pour le japonais, japonais prodrop etc. Le tableau 2.4-6 rapporte trois scores obtenus avec trois dispositifs de TAS. Même en tenant compte des nombreux biais de l'évaluation, qui minorent les résultats (faible qualité et quantité des corpus ja>fr ; particularité linguistique de ces corpus ; désavantage de l'évaluation du français avec BLEU), force est de constater que les rares chiffres disponibles pour ja>fr étaient sensiblement en deçà des valeurs obtenues pour le couple ja>an, *a fortiori* lorsque la traduction était complétée par un pré et post traitement linguistique.

	Dispositif	langues	corpus	BLEU
1	Google	ja>fr	OPUS-jafr)	$\pm$ 6.50
2	GIZA++ Moses 3.0	ja>fr	OPUS-jafr)	$\pm$ 13
3	Google	ja>fr	Titres presse	(veille) 6.6 / (fluide) 4.5
4	Giza++ Moses	ja>an	WikiKyoto	17.75
5	S2S + réordonnement	ja>an	WikiKyoto	19.35
6	Tree to string	ja>an	WikiKyoto	23.97

TABLE 2 – Performances de systèmes de TAS, en 2013 et 2015 ; (4) baseline académique chaîne à chaîne standard (Trieu *et al.*, 2017) et (5) avec réordonnement préalable des mots (Neubig *et al.*, 2012) ; (6) Traduction d'arbre à chaîne (Neubig & Duh, 2014)

.

Globalement, le passage au traitement neuronal voit une amélioration sensible des résultats pour la traduction ja>an. Parmi les nombreux chiffres disponibles, mentionons par exemple BLEU=26.22 (Cromieres *et al.*, 2016) contre 20.36 pour la TAS (Nakazawa *et al.*, 2015) sur un corpus de grande taille de résumés d'articles scientifiques (ASPEC-JE, (Nakazawa *et al.*, 2016)). On observe malgré tout des contre-performances, comme le score obtenu par (Trieu *et al.*, 2017), inférieur à celui de la TAS (Tableau 3-5). Les auteurs apportent une explication : la faible taille de WikiKyoto nuit à la qualité de la traduction. Pour un petit corpus (300K), la TAN serait désavantagé. Au delà, la première l'emporte. Cette conclusion ne saurait être toutefois définitive car nous voyons (5) qu'il est possible d'obtenir avec un dispositif standard de TAN des résultats équivalents à ceux obtenus avec un dispositif optimisé de TAS.

Les évaluations quantitatives du Tableau 3 montrent que la TAN n'a pas comblé le fossé entre TA ja>an et ja>fr. La TAN *directe* ja>fr progresse à peine (globalement, entre 1 et 2 points). Quant aux titres de presse, ils constituent un sous-langage du japonais qui reste difficile à traduire automatiquement en français. Le seul progrès sensible (4 points) a lieu grâce à la traduction indirecte, via l'anglais. Cela contredit (Blin, 2014) qui supposait que cette procédure handicapait les traductions. Mais cette contradiction n'est peut-être qu'apparente. Elle peut s'expliquer par la taille des corpus : les paires ja>an et an>fr sont richement dotées en corpus. Le gain en qualité pour chaque paire de langue suffirait à compenser le surcroît d'erreurs dû à la double traduction.

Nous avons complété les observations par un test «qualitatif» sur l'article du journal Asahi. Le principe est de compter le nombre d'actions basiques (élimination, ajout, déplacement,modification) à appliquer à chaque phrase traduite par le système pour obtenir une phrase de niveau «veille». Google

	Dispositif	langues	Corpus	Train.	Test	BLEU(v13)
1	Bing Transl.	ja > fr	2014		382	5.00 / 3.87
2	Goo. Transl.	ja > fr	2014		382	8.68 / 4.81
3	Goo. Transl.	ja > an > fr	2014		382	10.03 / 5.62
4	Goo. Transl.	ja>fr	AlignJaFr 0.5	91K	1 973	9.29
5	(Trieu *et al.*, 2017)	ja>an	WikiKyoto	389K	1 160	14,91
6	Goo. Transl.	ja>an	WikiKyoto		1 160	11.14

TABLE 3 – Performances de systèmes TAN avec japonais pour langue source.

Translate (tableau 4) s'en sort le mieux. Là encore, la traduction via l'anglais surpasse les autres et répond aux exigences d'une «veille» : la thématique est parfaitement compréhensible et les liens logiques sont pour la plupart rendus correctement. Bing semble procéder à une traduction via l'anglais. Le résultat permet de se faire une idée du thème, mais contient trop de contresens pour permettre de comprendre les liens logiques. Enfin Baidu Translator traduit via le chinois. La traduction est trop faible pour être exploitable. Sachant que les paires ja-zh et zh-fr sont peu et très peu dotées en corpus (Chen *et al.*, 2014), ces résultats confirment que la traduction par interlangue n'est profitable que si chaque paire de langue exploitée dispose de bonnes ressources.

	Dispositif	Langue	Ajouter	Effacer	Modifier	Déplacer	Total
1	Google Transl.	ja > fr	9	8	23	7	47
2	Google Transl.	ja > an > fr	8	6	16	8	38
3	Bing	ja > an > fr	16	6	23	9	54
4	Baidu	ja > zh > fr	Phrases à réécrire entièrement				

TABLE 4 – Evaluation manuelle de TAN sur un article de presse court.

3 Evaluation d'un baseline avec corpus

Une évolution remarquable de ces dernières année en TA, est la mise à disposition de nombreux baselines «clef en main». Nous observons l'un d'entre eux, OpenNMT-lua (abr. OpenNMT) (Klein *et al.*, 2017). Il a été choisi car les résultats obtenus sont corrects, selon les démonstrations faites par les auteurs, et comme nous le constatons nous-même dans les lignes qui suivent. Nous avons utilisé les réglages par défaut.

Pour la traduction ja>an, le dispositif, même sans optimisation, surpasse les deux dispositifs de TAN évalués sur le même corpus (Tableau 3-5,6), ainsi qu'un baseline TAS «basique» (Tableau 2-4). Par contre, il reste un peu en deça des système de TAS optimisés (Tableau 2-5,6). La traduction ja>fr dépasse la traduction ja>an (Tableau 5-2 *vs* Tableau 3). Ce dernier résultat doit être cependant relativisé. Le corpus AlignJaFr contient de nombreux segments brefs, qui peuvent faciliter la traduction. D'ailleurs, tous les autres résultats sont très inférieurs à l'existant : les traductions des titres de presse et de l'article du Asahi sont trop faibles pour être exploitables. Une explication possible est que le vocabulaire présent dans ces deux corpus est peu représenté dans le corpus d'entrainement. Le dispositif fonctionne donc bien seulement sur des corpus test représentatifs des corpus sur lesquels il a été entraîné.

	Lang.	Corpus	Test	BLEU
1	ja>an	WikiKyoto		19,07
2	ja>fr	AlignJaFr 0.5		20
3	ja>fr	AlignJaFrtitres 0.1		3 et 2
4	ja>fr	AlignJaFr 0.5	Article Asahi	inexploitable

TABLE 5 – Evaluations des traductions de OpenNMT-lua.

4 Bilan et perspectives

Les résultats ci-dessus ne font pas apparaître d'argument fort et clair en faveur d'une bascule vers la TAN pour traduire du japonais vers le français. Par contre, un faisceau d'indices vont dans ce sens. Le premier indice est la progression observée sur les traductions japonais>anglais. Ces progrès sont encourageants dans la mesure où anglais et français partagent des caractéristiques linguistiques qui les opposent de la même manière au japonais. Ce qui profite à l'un devrait profiter à l'autre. La progression des scores par rapport à la TAS, même si elle est souvent faible, constitue un deuxième indice. Troisièmement, le «baseline» obtient plusieurs scores «corrects». Le dernier indice est la qualité de la traduction de l'article de presse.

Grâce aux pré- et post-traitements, les système TAS ont pu progresser de quelques points (tab. 2-5,6). Des tests doivent être menés avec les systèmes de TAN pour comparer les effets de différents lemmatiseurs (Kytea, mecab, JUMAN) et dictionnaires (mecadic, unidic etc.). Une source de difficultés pour la traduction japonais (SOV, pro-drop, etc.) > français (SVO) est la différence d'ordre des mots. Conformément aux conclusions de (Du & Way, 2017), il sera profitable d'introduire des données linguistiques dans les corpus, sans pratiquer de réordonnement, qui donne de bons résultats pour la TAS mais pas la TAN.

Comme auparavant pour la TAS, la TAN du japonais vers le français est fortement handicapée par l'absence de corpus de grande taille. Rien ne permet de penser que cette situation évoluera à court et moyen terme. Il est donc nécessaire de contourner le problème pour progresser. Plusieurs solutions existent. La première consiste à traduire via l'anglais comme interlingua. Les résultats ont montré que cette technique était payante, même s'il reste à l'évaluer sur différents type de textes. La seconde solution serait de traduire les corpus eux-mêmes : profiter du grand corpus ja>an, et de l'assez bonne TAN de l'anglais vers le français. Enfin, la TAN elle-même offre de nouvelles solutions, notables par leur simplicité, comme par exemple l'usage de corpus multilingues (Johnson *et al.*, 2016). Le principe serait d'entraîner le modèle sur le corpus ja-fr et le corpus ja-an, ce qui permettrait de profiter de leur taille et de leur qualité. Néanmoins, toutes ces solutions vont dans le même sens du gonflement du corpus, ce qui n'est pas sans poser un problème de fond. Faute de ne pas disposer de connaissances sur le ratio progrès/taille du corpus, il n'est pas possible de prédire objectivement quelle quantité de corpus sera nécessaire pour obtenir le résultat escompté.

Enfin, pour améliorer la traduction des textes comme les titres de presse, il reste à explorer l'hybridation de la TAN avec la traduction par règles. En effet, les titres de presse sont des «textes» très brefs. De ce fait, il n'y a pas à gérer de liens à distance et la richesse syntaxique est mécaniquement limitée. Par contre, le vocabulaire est très riche. Ce type de texte est donc favorable à un traitement par règle, au moins partiellement. Il reste à définir les modalités de la combinaison traitement neuronal et par règle, en s'inspirant éventuellement de la technique utilisée pour traduire les dépêches économiques (Uchino *et al.*, 2001).

Remerciements

Je remercie Jean Bazantay (INALCO) pour les informations fournies à propos de l'évaluation manuelle des traductions.

Références

BLIN R. (2014). Evaluation des traductions automatiques en français des titres de presse japonais. *https ://hal.archives-ouvertes.fr/hal-01062005*.

BLIN R. (2018). Automatic Evaluation of Alignments without using a Gold-Corpus - Example with French-Japanese Aligned Corpora.

CHEN Y., WANG L., BOITET C. & SHI X. (2014). On-going Cooperative Research towards Developing Economy-Oriented Chinese-French SMT Systems with a New SMT Framework. In *Actes de la 21e conférence sur le Traitement Automatique des Langues Naturelles*, p. 401–406, Marseille, France : Association pour le Traitement Automatique des Langues. 19063.

CROMIERES F., CHU C., NAKAZAWA T. & KUROHASHI S. (2016). Kyoto University Participation to WAT 2016. In *Proceedings of the 3rd Workshop on Asian Translation (WAT2016)*, p. 166–174, Osaka, Japan : The COLING 2016 Organizing Committee.

DU J. & WAY A. (2017). Pre-reordering for Neural Machine Translation : Helpful or Harmful ? *The Prague Bulletin of Mathematical Linguistics*, **108**, 171–182.

JOHNSON M., SCHUSTER M., LE Q. V., KRIKUN M., WU Y., CHEN Z., THORAT N., VIÉGAS F. B., WATTENBERG M., CORRADO G., HUGHES M. & DEAN J. (2016). Google's Multilingual Neural Machine Translation System : Enabling Zero-Shot Translation. *CoRR*, **abs/1611.04558**.

KLEIN G., KIM Y., DENG Y., SENELLART J. & RUSH A. (2017). Opennmt : Open-Source Toolkit for Neural Machine Translation. In *Proceedings of ACL 2017, System Demonstrations*, p. 67–72, Vancouver, Canada : Association for Computational Linguistics.

NAKAZAWA T., MINO H., GOTO I., NEUBIG G., KUROHASHI S. & SUMITA E. (2015). Overview of the 2nd Workshop on Asian Translation. In *Proceedings of the 2nd Workshop on Asian Translation (WAT2015)*, p. 1–28, Kyoto, Japon. 00000.

NAKAZAWA T., YAGUCHI M., UCHIMOTO K., UTIYAMA M., SUMITA E., KUROHASHI S. & ISAHARA H. (2016). ASPEC : Asian Scientific Paper Excerpt Corpus. In N. C. C. CHAIR), K. CHOUKRI, T. DECLERCK, S. GOGGI, M. GROBELNIK, B. MAEGAARD, J. MARIANI, H. MAZO, A. MORENO, J. ODIJK & S. PIPERIDIS, Eds., *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, Paris, France : European Language Resources Association (ELRA).

NEUBIG G. & DUH K. (2014). On the Elements of an Accurate Tree-to-String Machine Translation System. In *The 52nd Annual Meeting of the Association for Computational Linguistics*, Baltimore, USA.

NEUBIG G., NAKATA Y. & MORI S. (2011). Pointwise Prediction for Robust, Adaptable Japanese Morphological Analysis. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics : Human Language Technologies : Short Papers - Volume 2*, HLT '11, p. 529–533, Stroudsburg, PA, USA : Association for Computational Linguistics.

NEUBIG G., WATANABE T. & MORI S. (2012). Inducing a Discriminative Parser to Optimize Machine Translation Reordering. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, p. 843–853, Jeju Island, Korea : Association for Computational Linguistics. 00045.

PAPINENI K., ROUKOS S., WARD T. & ZHU W.-J. (2002). BLEU : a Method for Automatic Evaluation of Machine Translation. In *Proc. ACL*, p. 311–318.

TAKEZAWA T., SUMITA E., SUGAYA F., YAMAMOTO H. & YAMAMOTO S. (2002). Toward a Broad-Coverage Bilingual Corpus for Speech Translation of Travel Conversations in the Real World. In *International Conference on Language Resources and Evaluation*, p. 147–152.

TIEDEMANN J. & NYGAARD L. (2004). The OPUS corpus - parallel & free. In *Proceedings of the Fourth International Conference on Language Resources and Evaluation*, Lisbon, Portugal. 00000.

TRIEU H.-L., TRAN D.-V. & NGUYEN L.-M. (2017). Investigating Phrase-Based and Neural-Based Machine Translation on Low-Resource Settings. Cebu Philippines. 00000.

UCHINO H., SHIRAI S., YOKOO A., OOYAMA Y. & FURUSE O. (2001). ALTFLASH : A Japanese-English Machine Translation System for Market Flash Reports. *The transactions of the Institute of Electronics, Information and Communication Engineers. D-II*, **84**(6), 1167–1174.

WU Y., SCHUSTER M., CHEN Z., LE Q. V., NOROUZI M., MACHEREY W., KRIKUN M., CAO Y., GAO Q., MACHEREY K., KLINGNER J., SHAH A., JOHNSON M., LIU X., KAISER L., GOUWS S., KATO Y., KUDO T., KAZAWA H., STEVENS K., KURIAN G., PATIL N., WANG W., YOUNG C., SMITH J., RIESA J., RUDNICK A., VINYALS O., CORRADO G., HUGHES M. & DEAN J. (2016). Google's Neural Machine Translation System : Bridging the Gap between Human and Machine Translation. *CoRR*, **abs/1609.08144**.

Des pseudo-sens pour améliorer l'extraction de synonymes à partir de plongements lexicaux

Olivier Ferret
CEA, LIST, Laboratoire Vision et Ingénierie des Contenus, Gif-sur-Yvette, F-91191 France.
olivier.ferret@cea.fr

RÉSUMÉ

Au-delà des modèles destinés à construire des plongements lexicaux à partir de corpus, des méthodes de spécialisation de ces représentations selon différentes orientations ont été proposées. Une part importante d'entre elles repose sur l'utilisation de connaissances externes. Dans cet article, nous proposons *Pseudofit*, une nouvelle méthode de spécialisation de plongements lexicaux focalisée sur la similarité sémantique et opérant sans connaissances externes. *Pseudofit* s'appuie sur la notion de pseudo-sens afin d'obtenir plusieurs représentations pour un même mot et utilise cette pluralité pour rendre plus génériques les plongements initiaux. Nous illustrons l'intérêt de *Pseudofit* pour l'extraction de synonymes et nous explorons dans ce cadre différentes variantes visant à en améliorer les résultats.

ABSTRACT

Pseudo-senses for improving the extraction of synonyms from word embeddings

Beyond the models for building word embeddings, some methods has been proposed for specializing such representations according to a particular perspective. Most of these methods rely on external knowledge. In this article, we propose *Pseudofit*, a new method for specializing word embeddings according to semantic similarity without any external knowledge. *Pseudofit* first exploits the notion of pseudo-sens for building several representations for a word and then uses these representations for making the initial embeddings more generic. We illustrate the interest of *Pseudofit* for acquiring synonyms et we study several variants and extensions of *Pseudofit* according to this perspective.

MOTS-CLÉS : Sémantique distributionnelle, spécialisation de plongements lexicaux.

KEYWORDS: Distributional semantics, specialization of word embeddings.

1 Introduction

Le développement des modèles neuronaux dans le domaine du Traitement Automatique des Langues s'est accompagné d'une attention spécifique accordée aux représentations qu'ils construisent et manipulent, en particulier celles relatives aux mots, appelées également plongements lexicaux (*word embeddings*). Cette attention s'est traduite par la définition de modèles permettant de construire de telles représentations à partir de larges corpus selon des critères distributionnels, à l'instar des modèles CBOW et Skip-gram (Mikolov *et al.*, 2013) ou de GloVe (Pennington *et al.*, 2014).

La popularisation de ces méthodes s'est rapidement accompagnée de travaux visant à modifier

a posteriori les représentations construites afin de leur donner une orientation spécifique. Cette orientation se matérialisant le plus souvent sous la forme de relations lexicales, les plongements lexicaux sont adaptés afin que les similarités qu'ils entretiennent reflètent les relations lexicales considérées. Kiela *et al.* (2015) biaisent ainsi des plongements lexicaux vers la notion de similarité ou de proximité sémantique en s'appuyant soit sur des synonymes, soit sur des associations lexicales libres. Des méthodes telles que Retrofitting (Faruqui *et al.*, 2015), Counter-fitting (Mrkšić *et al.*, 2016) ou PARAGRAM (Wieting *et al.*, 2015) s'inscrivent dans la même perspective.

Les méthodes opérant sans connaissances externes sont le plus souvent intrinsèquement liées à la façon dont les plongements lexicaux sont construits. Levy & Goldberg (2014) orientent ainsi des plongements vers la similarité sémantique plutôt que la proximité en utilisant des dépendances syntaxiques dans les contextes plutôt que des cooccurrents graphiques. Certaines méthodes *a posteriori* existent mais n'affichent pas d'orientation spécifique autre que d'améliorer les plongements de façon générale : la méthode All-but-the-Top (Mu, 2018), axée sur la réduction de la dimensionnalité des plongements, est de celles-ci tandis que (Vulić *et al.*, 2017), en exploitant des relations morphologiques extraites à partir de règles, se classe parmi les approches pauvres en besoins de connaissances.

Dans cet article, nous proposons *Pseudofit*, une méthode d'amélioration de plongements lexicaux n'utilisant pas de connaissances externes et visant la similarité sémantique en général et l'extraction de synonymes en particulier. L'idée est d'exploiter la notion de pseudo-sens issue de la désambiguïsation sémantique pour matérialiser les variations observables dans les représentations distributionnelles des mots et les prendre en compte explicitement pour créer des représentations dépassant ces variations. Nous montrons plus particulièrement l'intérêt de *Pseudofit* dans le cadre d'une évaluation intrinsèque et pour l'extraction de synonymes (cf. section 3.1). Nous testons également différentes variantes de la méthode permettant pour certaines d'entre elles de l'améliorer (cf. section 3.2).

2 Méthode

2.1 Principes

La représentation distributionnelle d'un mot, au sens des contextes distributionnels, est par essence construite à partir d'un corpus et varie donc d'un corpus à un autre. Cette variabilité existe aussi au sein d'un corpus, même si celui-ci est assez homogène : les représentations distributionnelles des mots construites à partir de chacune des deux moitiés d'un corpus ne sont ainsi pas identiques. Du point de vue d'une modélisation générale du sens des mots et en supposant que les différents sens d'un mot sont répartis de façon homogène entre ces deux moitiés de corpus, il est en revanche naturel de considérer que ces représentations devraient être identiques, ou au moins très proches, et que les différences observées sont pour l'essentiel contingentes. Dans cette optique, une représentation issue de la convergence des représentations construites à partir de chacun des deux sous-corpus devrait avoir un caractère plus générique et donc améliorer ses propriétés du point de vue de la similarité sémantique.

Une mise en œuvre très littérale de cette approche se heurte néanmoins à une difficulté notable : la qualité des représentations distributionnelles est très directement liée à la taille des corpus dont elles sont issues. Une approche conduisant à réduire la taille de ces corpus se traduit donc mécaniquement par une dégradation de qualité que ne compense pas en pratique l'amélioration induite par la convergence des représentations. Pour dépasser ce problème, la méthode *Pseudofit* que nous

proposons s'appuie sur la notion de pseudo-sens. Cette notion est à mettre en relation avec celle de pseudo-mot, introduite en désambiguïsation sémantique par Gale *et al.* (1992) et Schütze (1992). Un pseudo-mot est un mot artificiel constitué du regroupement de deux mots différents, chacun d'entre eux constituant un pseudo-sens de ce mot artificiel. *Pseudofit* adopte la perspective inverse en divisant arbitrairement les occurrences d'un même mot m, des noms dans le cas présent, en deux sous-ensembles : les occurrences de l'un sont étiquetées m_1 et les occurrences de l'autre, m_2. Chaque sous-ensemble constitue alors un pseudo-sens de ce mot. Une représentation distributionnelle est construite à la fois pour m, m_1 et m_2 selon les mêmes modalités, dans le cas présent au moyen d'un modèle neuronal produisant des plongements lexicaux.

La seconde phase de *Pseudofit* consiste à modifier *a posteriori* la représentation de m afin de tenir compte de la généralisation issue du rapprochement des représentations de m_1 et m_2. Considérer simultanément la représentation de m et celles de m_1 et m_2 permet à la fois de bénéficier de la qualité de la représentation de m, construite avec tout le corpus, et des différences des représentations de m_1 et m_2. Cette modification *a posteriori* est quant à elle réalisée en appliquant une méthode de spécialisation de plongements sur la base des relations de similarité sémantique existant par principe entre m, m_1 et m_2.

2.2 Construction des plongements lexicaux

La première étape de la méthode *Pseudofit* consiste donc à construire une représentation distributionnelle de chaque mot du corpus considéré ainsi que des deux pseudo-sens de ce mot. Le point de départ de cette construction est la génération pour chaque occurrence d'un mot d'un ensemble de contextes distributionnels. Classiquement, ces contextes sont générés à partir d'une fenêtre graphique centrée sur le mot cible. La particularité dans le cas présent est que ces contextes sont générés à la fois pour le mot cible et pour l'un de ses deux pseudo-sens. Le pseudo-sens concerné change d'une occurrence du mot à l'autre, conduisant ainsi au même nombre d'occurrences pour les deux pseudo-sens (à 1 occurrence près). Le tableau suivant offre une illustration des contextes générés pour les noms d'une phrase avec une fenêtre de trois mots (avant et après le mot cible) :

A policeman$_1$ was arrested by another policeman$_2$.

CIBLE	CONTEXTE	CIBLE	CONTEXTE	CIBLE	CONTEXTE
policeman	a	policeman$_1$	a	policeman$_2$	another
policeman	be	policeman$_1$	be	policeman$_2$	by
policeman	arrest (x2)	policeman$_1$	arrest	policeman$_2$	arrest
policeman	by (x2)	policeman$_1$	by		
policeman	another				

Cette phrase, au caractère volontairement artificiel, permet de montrer comment, à partir d'un même corpus, trois représentations différentes du même mot sont construites : l'une, à la première colonne, est construite à partir des contextes de toutes les occurrences du mot cible ; une deuxième, à la troisième colonne, est construite à partir d'une moitié des occurrences du mot cible, représentant son premier pseudo-sens, tandis que la troisième, à la cinquième colonne, s'appuie sur l'autre moitié des occurrences du mot cible, représentant son second pseudo-sens.

Compte tenu de la seconde phase de *Pseudofit*, les représentations distributionnelles construites sont des plongements lexicaux, produits ici grâce au modèle Skip-gram. Plus précisément, il s'agit de la

variante du modèle Skip-gram proposée par Levy & Goldberg (2014) et permettant de considérer des contextes arbitraires, formés de dépendances syntaxiques dans leur cas. Il est à noter que l'approche que nous proposons ne se limite pas au modèle Skip-gram puisque le même type de variante a été défini dans (Li *et al.*, 2017) pour les modèles CBOW et GloVe par exemple.

2.3 Convergence des représentations d'un même mot

L'objectif de la seconde phase est de faire converger les trois représentations de chaque mot cible obtenues lors de la première phase – la représentation du mot en tant que tel et celle de ses deux pseudo-sens – en faisant l'hypothèse qu'une telle convergence permet d'en construire une représentation plus générale en gommant des différences qui sont essentiellement contingentes puisque ces trois représentations font référence par principe à une même entité.

Cette convergence est mise en œuvre grâce à l'algorithme PARAGRAM. Celui-ci prend en entrée des plongements lexicaux et un ensemble de relations binaires entre mots caractérisant une forme de similarité sémantique. Ces relations sont supposées fournies a priori et sont généralement issues de réseaux lexicaux construits manuellement. PARAGRAM modifie graduellement les plongements en entrée par descente de gradient stochastique afin de rapprocher les vecteurs des mots impliqués dans une relation donnée. Ce rapprochement s'effectue sous la contrainte de ne pas trop s'éloigner des plongements initiaux. Ce double objectif se traduit plus formellement par la minimisation de la fonction objectif suivante, dans laquelle le rapprochement des vecteurs en fonction des relations données est porté par la première somme tandis que la seconde, modulée par le paramètre λ exprime le terme de conservation des plongements :

$$\underbrace{\sum_{(x_1,x_2)\in\mathcal{L}_i} \max\left(0, \delta + \mathbf{x}_1\mathbf{t}_1 - \mathbf{x}_1\mathbf{x}_2\right) + \max\left(0, \delta + \mathbf{x}_2\mathbf{t}_2 - \mathbf{x}_1\mathbf{x}_2\right)}_{\text{rapprochement des plongements en fonction des relations}} + \lambda \underbrace{\sum_{\mathbf{x}_i\in V(\mathcal{L}_i)} \left\|\mathbf{x}_i^{init} - \mathbf{x}_i\right\|^2}_{\text{conservation des plongements initiaux}}$$

L'idée directrice de PARAGRAM est de rapprocher les vecteurs des mots impliqués dans une relation donnée (maximisation du terme $\mathbf{x}_1\mathbf{x}_2$) tout en les éloignant des vecteurs des mots ne faisant pas partie de cette relation (minimisation des termes $\mathbf{x}_1\mathbf{t}_1$ et $\mathbf{x}_2\mathbf{t}_2$). Pour chaque mot $\mathbf{x}_i$ [1] d'une relation, un mot extérieur $\mathbf{t}_i$ à la relation est ainsi sélectionné comme point de référence pour cet éloignement. L'ensemble des relations étant divisé en mini-lots $\mathcal{L}_i$, la recherche de cette référence externe se limite aux mots du mini-lot ($V(\mathcal{L}_i)$) contenant la relation courante et maximise son pouvoir discriminant en sélectionnant le mot le plus proche, au sens de la mesure *Cosinus*, par rapport au mot considéré de la relation courante. Le paramètre δ permet de fixer la marge entre les termes de rapprochement et d'éloignement : l'objectif est que la similarité entre les mots $\mathbf{x}_1$ et $\mathbf{x}_2$ soit plus importante que leur similarité avec respectivement les mots $\mathbf{t}_1$ et $\mathbf{t}_2$, ceci avec une marge d'au moins δ.

PARAGRAM est appliqué aux plongements issus de la première phase avec l'objectif de rapprocher les représentations des mots et de leurs pseudo-sens. Pour chaque mot m, trois relations de similarité sont ainsi définies et injectées dans les plongements initiaux grâce à PARAGRAM : (m, m_1), (m, m_2) et (m_1, m_2). Au final, seules les représentations des mots sont utilisées et évaluées, étant de meilleure qualité puisque construites avec deux fois plus de données que celles des pseudo-sens.

1. $\mathbf{x}_i$ est plus précisément le vecteur représentant le plongement associé à un mot mais pour faciliter la présentation, nous assimilerons mot et vecteur dans ce qui suit.

3 Expérimentations

Pour mettre en œuvre et évaluer la méthode *Pseudofit*, nous avons sélectionné aléatoirement au niveau des phrases une sous-partie du corpus Annotated English Gigaword (Napoles *et al.*, 2012) formant un ensemble d'1 milliard de mots. Ce corpus est composé d'articles de journaux en anglais auxquels a été appliqué le Stanford CoreNLP toolkit (Manning *et al.*, 2014). Nous exploitons ce corpus sous une forme lemmatisée. Pour la construction des plongements lexicaux, nous avons utilisé l'outil *word2vecf*, adaptation de *word2vec* issue de (Levy & Goldberg, 2014), avec des paramètres tirés de (Baroni *et al.*, 2014) : fréquence minimale=5, taille vecteurs=300, taille fenêtre=5, 10 exemples négatifs et 10^{-5} pour le sous-échantillonnage des mots les plus fréquents. Pour les paramètres de PARAGRAM, nous avons adopté les valeurs préconisées dans (Vulić *et al.*, 2017), c'est-à-dire $\delta = 0,6$ et $\lambda = 10^{-9}$, avec AdaGrad (Duchi *et al.*, 2011) pour l'optimisation et 50 itérations.

3.1 Évaluation de *Pseudofit*

Notre première évaluation de *Pseudofit* est l'évaluation intrinsèque classiquement réalisée pour les plongements lexicaux. Elle consiste à calculer pour un ensemble de couples de mots la corrélation des rangs de Spearman ρ entre leurs similarités évaluées par un ensemble de personnes et celles calculées par la mesure *Cosinus* à partir des plongements à évaluer. Dans le cas présent, nous avons retenu des jeux de test de taille suffisante pour être significatifs en nous cantonnant aux noms. Il s'agit de SimLex-999 (Hill *et al.*, 2015), MEN (Bruni *et al.*, 2014) et MTurk-771 (Halawi *et al.*, 2012). Le tableau 1 compare les plongements produits par *Pseudofit* aux plongements initiaux ainsi qu'à ceux produits en remplaçant PARAGRAM par deux autres méthodes de spécialisation de plongements, Retrofitting et Counter-fitting. Pour les trois jeux de test, *Pseudofit* apporte une amélioration significative [2] par rapport à la condition initiale. En revanche, l'utilisation de Retrofitting ou de Counter-fitting pour remplacer PARAGRAM n'obtient pas une telle amélioration, voire conduit à une dégradation.

	SimLex-999	MEN	MTurk 771
INITIAL	0,495	0,783	0,656
Pseudofit	0,512	0,799	0,680
Retrofitting	0,496	0,774	0,650
Counter-fitting	0,495	0,772	0,649

TABLE 1: Évaluation intrinsèque de la méthode *Pseudofit*

La seconde évaluation que nous avons menée, objet principal de notre attention, possède un caractère plus extrinsèque. Il s'agit d'extraire les synonymes d'un mot. Pour ce faire, une mesure de similarité entre la représentation du mot cible et celle de chaque mot candidat est calculée, en l'occurrence la mesure *Cosinus*, et les candidats sont classés par ordre décroissant de leur valeur de similarité. Compte tenu de cet ordonnancement, nous avons adopté les mesures classiques en Recherche d'Information de R-précision (R-préc.), MAP (Mean Average Precision) et précision à différents rangs (P@r) pour nous comparer à notre référence, constituée par les synonymes de WordNet (Miller, 1990). Par ailleurs, nous avons adopté à la fois comme mots cibles et candidats les noms simples présents dans WordNet ayant plus de 10 occurrences dans chaque moitié de notre corpus, ce qui représente 20 813 noms.

2. La significativité statistique des différences a été évaluée par un test bilatéral de Steiger avec un seuil de significativité de 0,01, test implémenté grâce au module R *cocor* (Diedenhofen & Musch, 2015).

méthode	R-préc.	MAP	P@1	P@2	P@5	P@10
INITIAL	13,0	15,2	18,3	13,1	7,7	4,9
Pseudofit	+2,5	+3,3	+3,0	+2,5	+1,8	+1,1
Retrofitting	−0,5	−0,6	−0,6	−0,2†	−0,3	−0,2
Counter-fitting	−0,6	−0,8	−0,6	−0,5	−0,4	−0,3

TABLE 2: Évaluation de *Pseudofit* (résultats différentiels en points / INITIAL, x100)

Le tableau 2 donne le résultat de cette évaluation pour 11 481 noms ayant des synonymes dans WordNet parmi nos 20 813 cibles. Les plongements évalués sont les mêmes que ceux du tableau 1. Là encore, nous pouvons constater que *Pseudofit* se traduit par une amélioration significative des résultats [3] par rapport aux plongements initiaux. Par ailleurs, le remplacement de PARAGRAM par Retrofitting ou Counter-fitting se traduit dans ce cas par une dégradation notable des résultats. On peut faire l'hypothèse que l'ajout par PARAGRAM d'un terme d'éloignement par rapport aux mots non présents dans les relations de similarité contribue à renforcer le rapprochement des termes des relations, qui correspondent dans le cas présent aux différentes représentations d'un même mot que l'on souhaite justement voir converger. Il faut enfin ajouter, ce que ne montre pas le tableau 2 par manque de place, que *Pseudofit* est particulièrement efficace pour les mots de fréquences plus faibles (ici, la moitié inférieure des mots cibles en termes de fréquence) puisque pour ceux-ci, la R-précision augmente de 5,3 points, la MAP, de 6,7 points, la P@1, de 7,0 points et la P@2, de 5,2 points.

3.2 Variantes de *Pseudofit*

Nous avons testé plusieurs variantes de la méthode *Pseudofit*. La première, *Pseudofit max*, concerne la stratégie de sélection des $\mathbf{t}_i$ dans le cadre de PARAGRAM. Les résultats du tableau 1 ont été obtenus comme ceux de (Mrkšić *et al.*, 2017) dans une configuration où seule la moitié des $\mathbf{t}_i$ sont choisis en maximisant la similarité avec les $\mathbf{x}_i$, l'autre moitié étant choisie aléatoirement. Dans *Pseudofit max*, tous les $\mathbf{t}_i$ sont sélectionnés en fonction de leur similarité avec les $\mathbf{x}_i$. La deuxième variante, *Pseudofit 3 pseudo-sens*, cherche quant à elle à établir si une augmentation du nombre de pseudo-sens par mot peut influencer favorablement les résultats, en commençant par étendre ce nombre à trois. La troisième, *Pseudofit contexte*, teste si l'introduction de ces pseudo-sens au niveau des contextes distributionnels des mots présente un intérêt. Dans cette configuration, des pseudo-sens ont ainsi été distingués pour tous les noms, verbes et adjectifs suffisamment fréquents [4]. Enfin, la dernière variante, *Pseudofit fus-**, peut être vue elle-même comme une forme de variation de la deuxième. Elle introduit en effet une représentation supplémentaire du mot cible mais cette représentation, au lieu de prendre la forme d'un nouveau pseudo-sens, est constituée au contraire par une agrégation des représentations de ses deux pseudo-sens, ce qui reconstitue une forme de représentation globale de ce mot cible. Nous avons testé trois méthodes d'agrégation : *Pseudofit fus-addition* additionne les vecteurs des pseudo-sens dimension à dimension, *Pseudofit fus-moyenne* les moyenne tandis que *Pseudofit fus-max-pooling* en prend la valeur maximale.

Même si chaque variante se traduit par une amélioration des résultats par rapport à la méthode

3. La significativité statistique des différences a été évaluée grâce à un test de Wilcoxon pour échantillons appariés avec les notations suivantes : rien si $p \leq 0,01$, † si $0,01 < p \leq 0,05$ et ‡ si $p > 0,05$.

4. C'est-à-dire, les noms, verbes et adjectifs présents plus de 21 fois dans le corpus. Le seuil de 21 correspond à une fréquence minimale de 10 dans chaque moitié du corpus.

Variante	R-préc.	MAP	P@1	P@2	P@5	P@10
Pseudofit	15,5	18,5	21,3	15,6	9,5	6,0
Pseudofit max	+0,2‡	+0,3	+0,3†	+0,2†	+0,1	+0,1
Pseudofit 3 pseudo-sens	+0,2‡	+0,2	+0,4†	+0,2‡	+0,0‡	+0,1‡
Pseudofit contexte	+0,4†	+0,3‡	+0,5†	+0,2‡	+0,0‡	+0,0‡
Pseudofit fus-moyenne	+0,2†	+0,3	+0,4	+0,2†	+0,1	+0,1
Pseudofit fus-addition	+0,0‡	+0,0	+0,2‡	+0,1‡	+0,1†	+0,0‡
Pseudofit fus-max-pooling	+0,2‡	+0,3	+0,4	+0,2	+0,2	+0,1
Pseudofit max,fus-max-pooling	+0,4	+0,5	+0,5	+0,4	+0,2	+0,2

TABLE 3: Évaluation de variantes de *Pseudofit* (résultats différentiels en points / *Pseudofit*, x100)

Pseudofit de base, le tableau 3 montre que toutes n'ont pas le même intérêt. En termes à la fois du niveau d'amélioration et de significativité, *Pseudofit max* et *Pseudofit fus-max-pooling* sont les variantes les plus intéressantes, ce qui nous a conduit à les associer. Le résultat de cette combinaison, *Pseudofit max,fus-max-pooling*, obtient les meilleurs résultats, avec une différence significative par rapport à *Pseudofit* pour toutes les mesures. Parmi les variantes *Pseudofit fus-**, *Pseudofit fus-max-pooling* et *Pseudofit fus-moyenne* sont proches et surclassent assez nettement *Pseudofit fus-addition*. Les résultats de *Pseudofit 3 pseudo-sens* montrent quant à eux que l'utilisation d'un plus grand nombre de pseudo-sens se heurte probablement à un problème de qualité de leur représentation résultant d'un nombre d'occurrences plus restreint. Ce même effet de fréquence, cette fois-ci au niveau des contextes, explique vraisemblablement l'impact très limité de l'introduction des pseudo-sens dans les contextes observé pour *Pseudofit contexte*.

4 Conclusion et perspectives

Dans cet article, nous avons proposé *Pseudofit*, une nouvelle méthode permettant d'améliorer des plongements lexicaux sans a priori sur leur mode de construction et sans faire appel à des connaissances externes. Les évaluations menées ont permis de montrer que cette méthode est capable d'améliorer de façon très significative des plongements lexicaux en les orientant vers une similarité sémantique propre à favoriser l'extraction de synonymes. Les évaluations menées ont en outre permis de mieux cerner l'influence de certains facteurs de la méthode et d'en proposer des extensions aboutissant à de meilleurs résultats. Dans le travail présenté, les principes sous-jacents à *Pseudofit*, en particulier la convergence de représentations différentes d'un même mot, n'ont été testés qu'au sein d'un même corpus. En conjonction avec les travaux sur la construction de méta-plongements lexicaux (Yin & Schütze, 2016), il serait intéressant de les appliquer à des représentations construites à partir de corpus différents, à l'image de (Mrkšić *et al.*, 2017) pour des langues différentes.

Remerciements

Ce travail a été partiellement financé par l'Agence Nationale de la Recherche dans le cadre du projet ANR-17-CE23-0001 ADDICTE (Analyse distributionnelle en domaine de spécialité).

Références

BARONI M., DINU G. & KRUSZEWSKI G. (2014). Don't count, predict! A systematic comparison of context-counting vs. context-predicting semantic vectors. In *52nd Annual Meeting of the Association for Computational Linguistics (ACL 2014)*, p. 238–247, Baltimore, Maryland.

BRUNI E., TRAM N., BARONI M. *et al.* (2014). Multimodal distributional semantics. *Journal of Artificial Intelligence Research*, **49**, 1–47.

DIEDENHOFEN B. & MUSCH J. (2015). cocor: A Comprehensive Solution for the Statistical Comparison of Correlations. *PLOS ONE*, **10**(4), 1–12.

DUCHI J., HAZAN E. & SINGER Y. (2011). Adaptive Subgradient Methods for Online Learning and Stochastic Optimization. *Journal of Machine Learning Research*, **12**, 2121–2159.

FARUQUI M., DODGE J., JAUHAR S. K., DYER C., HOVY E. & SMITH N. A. (2015). Retrofitting Word Vectors to Semantic Lexicons. In *NAACL HLT 2015*, p. 1606–1615, Denver, Colorado.

GALE W. A., CHURCH K. W. & YAROWSKY D. (1992). Work on statistical methods for word sense disambiguation. In *AAAI Fall Symposium on Probabilistic Approaches to Natural Language*, p. 54–60.

HALAWI G., DROR G., GABRILOVICH E. & KOREN Y. (2012). Large-scale Learning of Word Relatedness with Constraints. In *18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD'12)*, p. 1406–1414: ACM.

HILL F., REICHART R. & KORHONEN A. (2015). Simlex-999: Evaluating semantic models with (genuine) similarity estimation. *Computational Linguistics*, **41**(4), 665–695.

KIELA D., HILL F. & CLARK S. (2015). Specializing Word Embeddings for Similarity or Relatedness. In *2015 Conference on Empirical Methods in Natural Language Processing (EMNLP 2015)*, p. 2044–2048, Lisbon, Portugal.

LEVY O. & GOLDBERG Y. (2014). Dependency-Based Word Embeddings. In *52nd Annual Meeting of the Association for Computational Linguistics (ACL 2014)*, p. 302–308, Baltimore, Maryland.

LI B., LIU T., ZHAO Z., TANG B., DROZD A., ROGERS A. & DU X. (2017). Investigating Different Syntactic Context Types and Context Representations for Learning Word Embeddings. In *2017 Conference on Empirical Methods in Natural Language Processing (EMNLP 2017)*, p. 2421–2431, Copenhagen, Denmark.

MANNING C. D., SURDEANU M., BAUER J., FINKEL J., BETHARD S. J. & MCCLOSKY D. (2014). The Stanford CoreNLP Natural Language Processing Toolkit. In *52nd Annual Meeting of the Association for Computational Linguistics (ACL 2014), system demonstrations*, p. 55–60.

MIKOLOV T., CHEN K., CORRADO G. & DEAN J. (2013). Efficient estimation of word representations in vector space. In *ICLR 2013, workshop track*.

MILLER G. A. (1990). WordNet: An On-Line Lexical Database. *International Journal of Lexicography*, **3**(4).

MRKŠIĆ N., Ó SÉAGHDHA D., THOMSON B., GAŠIĆ M., ROJAS-BARAHONA L. M., SU P.-H., VANDYKE D., WEN T.-H. & YOUNG S. (2016). Counter-fitting Word Vectors to Linguistic Constraints. In *2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL HLT 2016)*, p. 142–148, San Diego, California.

MRKŠIĆ N., VULIĆ I., Ó SÉAGHDHA D., LEVIANT I., REICHART R., MILICA G., KORHONEN A. & YOUNG S. (2017). Semantic Specialization of Distributional Word Vector Spaces using

Monolingual and Cross-Lingual Constraints. *Transactions of the Association for Computational Linguistics*, **5**, 309–324.

MU J. (2018). All-but-the-Top: Simple and Effective Postprocessing for Word Representations. In *Sixth International Conference on Learning Representations (ICLR 2018), poster session*, Vancouver, Canada.

NAPOLES C., GORMLEY M. R. & VAN DURME B. (2012). Annotated Gigaword. In *NAACL Joint Workshop on Automatic Knowledge Base Construction and Web-scale Knowledge Extraction (AKBC-WEKEX)*, p. 95–100, Montréal, Canada.

PENNINGTON J., SOCHER R. & MANNING C. D. (2014). GloVe: Global Vectors for Word Representation. In *2014 Conference on Empirical Methods in Natural Language Processing (EMNLP 2014)*, p. 1532–1543, Doha, Qatar.

SCHÜTZE H. (1992). Dimensions of meaning. In *1992 ACM/IEEE conference on Supercomputing*, p. 787–796: IEEE Computer Society Press.

VULIĆ I., MRKŠIĆ N., REICHART R., Ó SÉAGHDHA D., YOUNG S. & KORHONEN A. (2017). Morph-fitting: Fine-Tuning Word Vector Spaces with Simple Language-Specific Rules. In *55th Annual Meeting of the Association for Computational Linguistics (ACL 2017)*, p. 56–68, Vancouver, Canada: Association for Computational Linguistics.

WIETING J., BANSAL M., GIMPEL K. & LIVESCU K. (2015). From Paraphrase Database to Compositional Paraphrase Model and Back. *Transactions of the Association for Computational Linguistics*, **3**, 345–358.

YIN W. & SCHÜTZE H. (2016). Learning Word Meta-Embeddings. In *54th Annual Meeting of the Association for Computational Linguistics (ACL 2016)*, p. 1351–1360, Berlin, Germany.

Annotation automatique des types de discours dans des livres audio en vue d'une oralisation par un système de synthèse

Aghilas Sini[1] Elisabeth Delais-Roussarie[2] Damien Lolive[1]
(1) Univ Rennes, CNRS, IRISA, 6 rue de Kerampont, 22300 Lannion, France
(2) UMR 6310 - Laboratoire de Linguistique de Nantes, 7 Chemin de la Censive du Tertre, 44312 Nantes, France
aghilas.sini@irisa.fr, elisabeth.delais-roussarie@univ-nantes.fr,
damien.lolive@irisa.fr

RÉSUMÉ

Pour synthétiser automatiquement et de manière expressive des livres audio, il est nécessaire de connaître le type des discours à oraliser. Ceci étant, dans un roman ou une nouvelle, les perspectives narratives et les types de discours évoluent souvent entre de la narration, du récitatif, du discours direct, du discours rapporté, voire des dialogues. Dans ce travail, nous allons présenter un outil qui a été développé à partir de l'analyse d'un corpus de livres audio (extraits de *Madame Bovary* et des *Mystères de Paris*) et qui prend comme unité de base pour l'analyse le paragraphe. Cet outil permet donc non seulement de déterminer automatiquement les types de discours (narration, discours direct, dialogue), et donc de savoir qui parle, mais également d'annoter l'extension des modifications discursives. Ce dernier point est important, notamment dans le cas d'incises de citation où le narrateur reprend la parole dans une séquence au discours direct. Dans sa forme actuelle, l'outil atteint un taux de 89 % de bonne détection.

ABSTRACT

Automatic annotation of discourse types in audio-books

To synthesize audiobooks in an expressive manner, it is necessary to know the type of discourses that have to be produced. However, in a novel or a tale, narrative perspectives and discourse types often change, moving from narrative and recitative paragraphs to direct speech, reported speech, and even dialogs. In this work, we will present a tool that was developed from the analysis of a corpus (including excerpts from *Madame Bovary* and *Les Mystères de Paris*) and that relies on paragraph as **basic unit**. It allows not only to automatically determine the type of speech (narrative speech, direct speech, dialogs), and therefore to know who is speaking, but also to annotate the extension of the discursive modifications. This later point is important, especially in the case of parentheticals with reporting verbs where the narrator speaks again in the middle of a direct speech sequence. In its current form, the tool achieves a 89 % detection rate.

MOTS-CLÉS : types de discours, discours direct, incises, annotation automatique.

KEYWORDS: discourse types, direct speech, quotation and reporting verb, automatic annotation.

1 Introduction

Pour synthétiser de façon satisfaisante et expressive des livres audio, il est important de pouvoir indiquer toute modification dans les perspectives énonciatives (les changements de locuteur dans les

séquences dialoguées, les incises de citation, etc.) par des marquages prosodiques comparables à ceux observés dans la parole naturelle (Doukhan *et al.*, 2011; Montaño *et al.*, 2013). Pour y parvenir, il est nécessaire de distinguer dans un texte les paragraphes (en entendant par paragraphe toute séquence séparée par des sauts de ligne dans le texte) selon leur type de discours, et de là d'avoir une idée précise de "qui parle". Cela conduit à classer les paragraphes selon qu'ils correspondent à des passages narratifs (1), dialogués (2), ou mixtes. De fait, dans certains paragraphes, du discours rapporté est inséré au milieu de passages narratifs (3) ou des incises de citation, correspondant à du discours narratif, apparaissent dans des discours directs, ces dernières pouvant être courtes (4a) ou relativement longues (4b). Dans ces cas mixtes, la tâche consiste à délimiter avec précision les types de discours en présence.

(1) On commença la récitation des leçons. Il les écouta de toutes ses oreilles, attentif comme au sermon, n'osant même croiser les cuisses, ni s'appuyer sur le coude, et, à deux heures, quand la cloche sonna, le maître d'études fut obligé de l'avertir, pour qu'il se mit avec nous dans les rangs. (*Madame Bovary*, chap. 10)

(2) – D'où viens-tu encore, gredin ?
– Vous êtes bien curieux, sans yeux... (*Les Mystères de Paris*, chap. 7, Partie 2)

(3) D'autre part, la mort de sa femme ne l'avait pas mal servi dans son métier, car on avait répété durant un mois : **« Ce pauvre jeune homme ! quel malheur ! »** (*Madame Bovary*, chap. 3)

(4) a. – Levez-vous, **reprit le professeur**, et dites-moi votre nom... (*Madame Bovary*, chap. 1)
b. – Débarrassez-vous donc de votre casque, **dit le professeur, qui était un homme d'esprit**. (*Madame Bovary*, chap. 1)

(5) Puis, l'ayant considéré quelques minutes d'un œil amoureux et tout humide, **elle dit vivement :** (*Madame Bovary*, chap. 18)

Alors que la détection des passages narratifs (1), des passages dialogués (2) et des discours rapportés aux milieu de passages narratifs (3) dans les paragraphes mixtes peut sembler assez triviale, du fait notamment d'indications typographiques, l'annotation des incises de citation est plus complexe, comme en témoigne une simple comparaison entre les cas (4a) et (4b). La présence d'une virgule après l'incise de citation n'est en effet pas suffisante.

Dans cet article, nous allons présenter l'outil que nous avons développé pour annoter automatiquement un texte et indiquer clairement les changements énonciatifs et discursifs. Dans un premier temps, nous présentons le corpus utilisé pour cette étude, et nous indiquons comment nous avons procédé pour délimiter les différents types de discours. Dans une seconde partie, nous fournissons des indications sur les taux d'identification obtenus et sur les problèmes résiduels.

Typologie des paragraphes	Discours direct	Discours indirect	Discours mixte
Paragraphes	1 202	844	771
Phrases	4 109	2 160	2 920
Mots	36 722	36 622	26 001
Mots orthographiquement distincts	5399	6913	4 345
Mots phonétiquement distincts	5235	6764	4 248
Syllabes	49 313	55 021	35 827
Syllabes différentes	2 692	2678	2 279
Phonèmes	111 915	124 886	80 827
Phonèmes distincts	33	33	33

TABLE 1 – Composition du corpus en fonction des types de discours

2 Corpus et procédure d'annotation des types de discours

2.1 Corpus et matériel

Pour cette étude, nous avons travaillé sur une sous-partie d'un corpus d'audiobooks comprenant 87 heures de lecture par une unique locutrice de plusieurs livres audio (Sini *et al.*, 2018). Ce dernier a été développé afin de travailler à l'amélioration de l'expressivité en synthèse par corpus et a été collecté depuis la librairie LibriVox [1]. Dans le travail présenté ici, nous avons retenu des extraits correspondants à des chapitres de deux romans français, *les Mystères de Paris* d'Eugène Sue et *Madame Bovary* de Gustave Flaubert, pour une durée totale de 10 heures. Ces extraits ont été choisis par un expert sur l'ensemble des deux œuvres. Ils ont été retenus car ils renferment de nombreux changements de perspectives discursives et énonciatives, tout en permettant d'arriver à un ensemble relativement cohérent en termes de séquences au discours direct et indirect comme indiqué dans le tableau 1.

Pour l'ensemble du corpus et de la sous-partie retenue pour cette étude, nous disposons de la transcription orthographique, de la phonétisation et de l'alignement sur le signal sonore fait automatiquement à l'aide de JTrans (Cerisara *et al.*, 2009). D'autres annotations linguistiques de nature phonologiques (découpage en syllabes, etc.) et morpho-syntaxiques (catégorisation grammaticale des mots, analyse et indication des fonctions grammaticales) sont également disponibles grâce au recours à des procédures d'annotation automatique (Candito *et al.*, 2010, 2009). L'ensemble du processus d'annotation a été mené en s'appuyant sur ROOTS (Chevelu *et al.*, 2014), ce qui permet de maintenir l'ensemble des annotations de façon cohérente.

2.2 Procédure d'annotation

L'annotation automatique des changements discursifs et énonciatifs pour un texte donné (un chapitre dans notre cas) se fait en deux phases, illustrées dans les sous-sections qui suivent :

1. https ://librivox.org

1. Classification des paragraphes en fonction des types de discours, de manière à conserver les paragraphes qui comportent des incises ;
2. Détection et délimitation des incises de citation (*dit-il*, etc.) et des amorces (*il affirma :"..."*, etc.).

2.2.1 Classification des paragraphes selon les types de discours

À partir du texte, le programme classe les paragraphes, définis sur une base typographique (passage à la ligne), en trois groupes distincts (voir fig. 1, phase 1). Il s'appuie pour cela sur des critères lexico-syntaxiques (présence de verbe de discours, etc.), ainsi que sur la ponctuation et des signes typographiques (présence de guillemets ou de tirets, etc.) :

— le groupe *Discours Direct* regroupe tous les paragraphes qui contiennent exclusivement des passages au discours direct comme dans l'exemple (2) ;
— le groupe *Narration* renferme les paragraphes ne comportant que des passages de narration ou des descriptions comme dans l'exemple (1) ;
— le groupe *Discours mixte* est composé de paragraphes qui peuvent contenir à la fois du discours direct et rapporté et du discours indirect ou de la narration. Seront présents dans ce groupe à la fois les passages narratifs dans lesquels sont insérés des discours rapportés comme dans l'exemple (3) et des passages au discours direct comprenant des incises de citation comme dans l'exemple (4).

Dans une seconde phase (voir fig. 1, phase 2), les paragraphes du groupe *Discours Mixte* sont analysés afin de déterminer les frontières exactes des changements de discours. Cette tâche est effectuée sur un mode expert par règles. Notons cependant que d'autres travaux ont eu recours à des techniques d'apprentissage automatique pour une tâche analogue (Schöch *et al.*, 2016). Cette étape va permettre d'identifier les passages au discours rapporté, souvent entre guillemets et précédés des deux points comme (3), mais aussi les incises de citation dans les passages dialogués comme (4), et les séquences amorces introduisant un passage au discours direct ou un dialogue comme (5). Parmi ces éléments, les incises de citation sont importantes car elles permettent de délimiter les changements de locuteur et de fournir des indications sur les personnages en présence et sur leur attitude.

2.2.2 Détection et annotation des incises et des amorces

À l'issue de la première phase de classification, les paragraphes *Discours mixte* sont analysés de façon détaillée pour déterminer les frontières des différents types de discours. La méthode implémentée pour détecter les incises de citation s'appuie dans un premier temps sur les travaux de (Boula de Mareüil & Maillebau, 2002), qui consiste en un ensemble d'expressions régulières. Puis on y ajoute un ensembles de règles qui visent à détecter les incises d'une manière plus détaillée, et à couvrir des cas plus complexe en s'appuyant sur l'analyse syntaxique des incises de citation décrit par (Bonami & Godard, 2008) et (Danlos *et al.*, 2010). Dans notre étude trois configurations ont été distinguées :

— les amorces de discours direct comme dans l'exemple (5) ;
— les incises de citation situées au milieu de la prise de parole d'un personnage (4a) ;
— les incises de citation placées à la fin des propos d'un personnage (4b).

S'ajoutent à ces trois configurations les cas où un discours direct est inséré dans un discours indirect

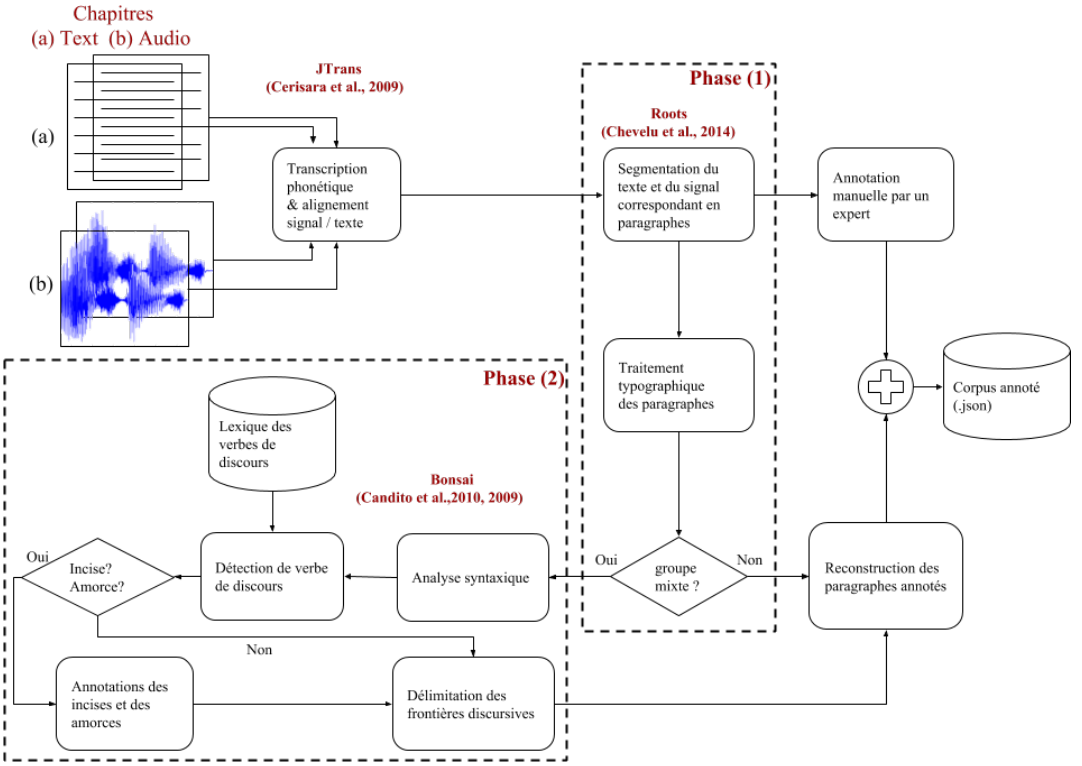


FIGURE 1 – Processus d'annotation des chapitres

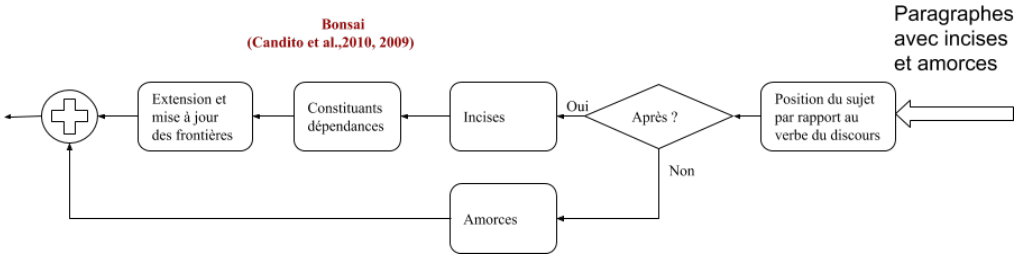


FIGURE 2 – Détection et annotation des incises et des amorces

de manière soudaine, c'est-à-dire sans amorce ou autre indication de changement de perspective discursive.

Pour analyser ces différentes configurations, il est nécessaire de regarder d'autres éléments que la seule ponctuation ou présence de tirets. Dans l'approche proposée, nous prenons en compte à la fois le résultat de l'analyseur syntaxique (Candito *et al.*, 2009) et un lexique de 327 verbes de discours (*affirmer, répéter, s'écrier, dire,* etc.). Lorsqu'un verbe de discours, généralement à la troisième personne du singulier (dans 97% des cas), est détecté, on s'intéresse à son sujet, afin de connaître sa position par rapport au verbe. Deux cas se présentent : si le sujet est à gauche du verbe (avant), il s'agit d'une amorce ; si, au contraire, il est après le verbe, on a affaire à une incise. Dans ce cas, il est important d'en établir l'extension, les incises pouvant être courtes (comme (4a)) ou relativement longue (voir (4b)). Pour ce faire, on s'appuie sur la ponctuation, mais également sur l'analyse syntaxique, notamment pour les éléments à droite du verbe qui peuvent dépendre du sujet comme dans le cas d'une apposition ou d'une relative (voir exemple (4b)). Le processus complet est illustré dans la fig. 2.

3 Résultats et analyse

Les résultats obtenus par cet algorithme de détection des types de discours sont donnés dans le tableau 2. Les performances ont été estimées avec trois mesures : la précision, le rappel, et la F-mesure.

L'algorithme permet une bonne classification des paragraphes (92,19 % de bonne détection ou F-mesure) à l'issue de la phase 1. Pour l'analyse des incises (801 annotées manuellement sur le corpus), les performances ont été calculées en distinguant deux niveaux d'annotation. La détection simplifiée - laquelle repose sur la prise en compte des verbes de discours et de la ponctuation (voir fig.1, phase 2) - ne permet pas de délimiter avec précision l'extension des incises (F-mesure : 86,3 %), des erreurs apparaissant lorsqu'une relative ou une apposition dépendent du sujet. La prise en compte de l'analyse syntaxique et des dépendances comme indiqué dans la figure 2 permet, en revanche, d'affiner les résultats et de les améliorer, si bien qu'on atteint, à l'issue de cette détection précise, un score de 89,09 % (ce dernier comprenant le type de discours et l'extension des incises).

	Précision	Rappel	F-mesure
Annotations des paragraphes (Phase 1)	92,6	91,2	92,19
Détection simplifiée des types de discours (direct, indirect, incises et amorces)	87,5	85,2	86,33
Détection précise des incises (avec délimitation fine)	89,7	88,5	89,09

TABLE 2 – Résultats de la détection et de l'annotation des changements discursifs

Une étude des erreurs permet d'isoler deux cas :

— ceux où le verbe de discours prend la forme d'un participe, et non d'un verbe conjugué, comme dans l'exemple (6). L'algorithme a en effet tendance à limiter l'incise à *arrêta*.

(6) – Cinq cents vers à toute la classe ! **exclamé d'une voix furieuse, arrêta, comme le Quos ego, une bourrasque nouvelle.** (*Madame Bovary, Chapitre 1*)

— ceux où l'analyse syntaxique effectuée est erronée comme dans l'extrait (7). La complexité et les enchâssements syntaxiques dans l'incise rendent son analyse difficile.

(7) – Oui... j'entends bien ; vous voulez que je vous mène à sa porte... et puis à son lit... et puis que je vous dise où frapper, et puis que je vous guide le bras, n'est-ce pas ? Vous voulez enfin me faire servir de manche à votre couteau !... vieux monstre ! **reprit Tortillard avec une expression de mépris, de colère et d'horreur qui, pour la première fois de la journée, rendit sérieuse sa figure de fouine, jusqu'alors railleuse et effrontée.** On me tuerait plutôt... entendez-vous... que de me forcer à vous conduire chez votre femme. (*Les Mystères de Paris,Chapitre 7, Partie 2*)

Les résultats obtenus sont moins bons que ceux présentés dans des travaux de classification des discours direct et indirect reposant sur des algorithmes d'apprentissage automatique (voir (Schöch *et al.*, 2016) qui obtient une F-mesure de 93.9 % en utilisant l'algorithme "Forêt d'arbres décisionnels"). Ceci étant, cette différence est à prendre avec précaution car les objectifs recherchés ne sont pas exactement les mêmes. La procédure développée par (Schöch *et al.*, 2016) vise à dire si chaque phrase est au discours direct ou indirect, mais n'isole pas les incises de citation, les amorces ou les passages au discours direct dans une séquence narrative. Cela s'explique par une différence fondamentale d'objectifs : alors que (Schöch *et al.*, 2016) veut classifier les œuvres sur des bases littéraires en s'appuyant sur la présence ou non de discours direct, nous souhaitons précisément savoir *qui parle* et à quel moment précis s'opère le changement. De plus, les différences de résultats peuvent s'expliquer par la méthode retenue : alors que (Schöch *et al.*, 2016) prend comme unité de base la phrase, nous prenons le paragraphe dans le but d'indiquer tout changement discursif dans un même paragraphe. En outre, nous avons recours à un modèle expert avec l'usage de règles, tandis qu'ils utilisent des procédures d'apprentissage automatique. Il serait d'ailleurs intéressant d'utiliser des procédures analogues à celles retenues par (Schöch *et al.*, 2016), mais en gardant les mêmes objectifs, à savoir déterminer précisément où s'opèrent les changements discursifs.

4 Conclusion et perspectives

Dans cet article, nous avons proposé un algorithme qui permet d'annoter automatiquement et avec précision les changements discursifs dans les livres audio. Les performances de l'outil sont relativement encourageantes, mais des erreurs subsistent dans les cas syntaxiquement complexes. Nous envisageons de nous appuyer sur le signal audio pour avoir une meilleure appréhension des changements discursifs en général, et pour mieux délimiter l'extension des incises dans les cas complexes. Cela pourrait ensuite servir de base pour une classification à l'aide d'outils d'apprentissage automatique.

5 Remerciements

Le travail présenté ici a été soutenu par l'opération PPC 7 du Labex "Empirical Foundations in Linguistics" (ANR-10-LABX-0083). Il a également bénéficié du soutien financier de l'Agence Nationale de la Recherche dans le cadre du projet ANR SynPaFlex (ANR-15-CE23-0015).

Références

BONAMI O. & GODARD D. (2008). Syntaxe des incises de citation. In *Actes du premier Congrès Mondial de Linguistique Française*, p. 2395–2408, France.

BOULA DE MAREÜIL P. & MAILLEBAU E. (2002). Traitement des incises en français : capture automatique et modèle prosodique. In *XXIVèmes Journées d'Étude sur la Parole, Nancy*.

CANDITO M., CRABBÉ B., DENIS P. & GUÉRIN F. (2009). Analyse syntaxique du français : des constituants aux dépendances. In *16e Conférence sur le Traitement Automatique des Langues Naturelles - TALN 2009*, Senlis, France.

CANDITO M., NIVRE J., DENIS P. & ANGUIANO E. H. (2010). Benchmarking of statistical dependency parsers for french. In *Proceedings of the 23rd International Conference on Computational Linguistics : Posters*, p. 108–116 : Association for Computational Linguistics.

CERISARA C., MELLA O. & FOHR D. (2009). Jtrans, an open-source software for semi-automatic text-to-speech alignment. In *Proceedings of the 10th Annual Conference of the International Speech Communication Association-Interspeech 2009*.

CHEVELU J., LECORVÉ G. & LOLIVE D. (2014). ROOTS : a toolkit for easy, fast and consistent processing of large sequential annotated data collections. In *Language Resources and Evaluation Conference (LREC)*, Reykjavik, Iceland.

DANLOS L., SAGOT B. & STERN R. (2010). Analyse discursive des incises de citation. In *2ème Congrès Mondial de Linguistique Française - CMLF 2010*, La Nouvelle Orléans, United States : Institut de Linguistique Française.

DOUKHAN D., RILLIARD A., ROSSET S., ADDA-DECKER M. & D'ALESSANDRO C. (2011). Prosodic Analysis of a Corpus of Tales. p. 3129–3132, Florence, Italy : International Speech Communication Association (ISCA).

MONTAÑO R., ALÍAS F. & FERRER J. (2013). Prosodic analysis of storytelling discourse modes and narrative situations oriented to text-to-speech synthesis. In *Eighth ISCA Workshop on Speech Synthesis*.

SCHÖCH C., SCHLÖR D., POPP S., BRUNNER A., HENNY U. & TELLO J. C. (2016). Straight talk ! automatic recognition of direct speech in nineteenth-century french novels. In *Digital Humanities 2016 : Conference Abstracts*, p. 346–353.

SINI A., LOLIVE D., VIDAL G., TAHON M. & ÉLISABETH DELAIS-ROUSSARIE (2018). Synpaflex-corpus : An expressive french audiobooks corpus dedicated to expressive speech synthesis. In *Language Resources and Evaluation Conference (LREC) to appear*.

Impact du Prétraitement Linguistique sur l'Analyse des Sentiments du Dialecte Tunisien

Chedi Bechikh Ali[1] Halla Mulki[2] Hatem Haddad[3]
(1) Institut Supérieur de Gestion, Tunis, Tunisie
(2) Département de génie informatique, Université Selcuk, Turquie
(3) Département d'informatique et d'ingénierie décisionnelle, Université Libre de Bruxelles, Belgique
chedi.bechikh@gmail.com, halamulki@selcuk.edu.tr, Hatem.Haddad@ulb.ac.be

RÉSUMÉ
Ce travail présente une étude de l'impact du prétraitement linguistique (suppression de mots vides, racinisation et détection d'emoji, de négation et d'entités nommées) sur la classification des sentiments en dialecte Tunisien. Nous évaluons cet impact sur trois corpus de tailles et contenus différents. Deux techniques de classification sont utilisées : Naïve bayes et Support Vector Machines. Nous comparons nos résultats aux résultats de référence obtenus sur ces même corpus. Nos résultats soulignent l'impact positif de la phase de prétraitement sur la performance de la classification.

This work presents a study of the impact of linguistic preprocessing (stop words elimination, stemming and detection of emoji, negation and named entities). We evaluate this impact on three datasets of different sizes and contents. Two classification techniques are used : Naive bayes and Support Vector Machines. We compare our results with the baselines results obtained from these same datasets. Our results highlight the positive impact of the preprocessing phase on the classification performance.

MOTS-CLÉS : Analyse de sentiment, dialecte tunisien, prétraitement de texte, entités nommées.

KEYWORDS: Tunisian sentiment analysis, text preprocessing, named entities.

1 Introduction

Les utilisateurs des réseaux sociaux ont tendance à utiliser un langage informel pour exprimer leurs opinions. A l'opposé de la langue arabe standard moderne, le langage arabe informel combine une variété de dialectes différents les uns des autres ; c'est pourquoi certains mots ou expressions peuvent exprimer des sentiments radicalement différents. Pendant et après la révolution tunisienne, le suivi des réactions et des opinions du public concernant les différents événements a été menée à travers des systèmes d'analyse des sentiments (Akaichi, 2014). Les travaux antérieurs sur l'analyse des sentiments (AS) du dialecte tunisien ont principalement traité les données textuelles en utilisant les procédures classiques de nettoyage et de normalisation (Sayadi *et al.*, 2016; Medhaffar *et al.*, 2017a; Karmani, 2017). Bien que ces modèles aient obtenu des résultats assez satisfaisants, l'amélioration de la classification des sentiments par l'application d'autres prétraitements reste un domaine de recherche intéressant. Une des motivations de cet article est l'exploitation de mots indicatifs de sentiments dérivés du corpus, tels que les entités nommées (EN), et leur inclusion dans l'étape de prétraitement peut contribuer à inférer le sentiment. En effet, les textes porteurs d'opinions sont riches d'entités nommées (personnes, lieux ou organisations) envers lesquels le sentiment est exprimé (Yasavur *et al.*,

2014). Nous supposons que la reconnaissance des entités nommées peut être exploitée dans l'analyse des sentiments si les entités nommées extraites sont classées sentimentalement comme porteuses d'opinion en fonction du contexte local dans lequel elles sont mentionnées. Au meilleur de notre connaissance, les entités nommées n'ont pas été utilisées dans des travaux antérieurs sur les systèmes d'AS du dialecte tunisien.

Dans cet article, nous cherchons à améliorer la performance de l'AS du dialecte tunisien par l'application unique ou combinée des prétraitements suivants : suppression des mots vides, racinisation, détection de négation et reconnaissance des emojis les plus utilisés. En outre, nous introduisons l'étiquetage des entités nommées en tant que prétraitement et nous étudions son impact sur les performances de la classification des sentiments lorsqu'il est combiné avec d'autres prétraitements. Pour évaluer notre approche, trois corpus tunisiens de tailles différentes fournis par (Sayadi *et al.*, 2016; Medhaffar *et al.*, 2017a; Karmani, 2017) et contenant des tweets positifs/négatifs et des commentaires sur plusieurs domaines ont été utilisés.

2 Analyse du sentiment du dialecte arabe

L'analyse des sentiments du dialecte tunisien peut être effectuée en utilisant des approches d'apprentissage automatique telles que des méthodes supervisées ou des approches basées sur le lexique.

- Méthode basée sur l'apprentissage supervisé : Cette méthode nécessite un corpus étiqueté pour entraîner le classifieur pour prédire la polarité du texte (Piryani *et al.*, 2017). Le processus d'apprentissage est réalisé en déduisant qu'une combinaison des caractéristiques spécifiques d'une phrase donne une classe de polarité spécifique : positive, négative. Les caractéristiques utilisées avec cette stratégie sont des caractéristiques en sac de n-grammes. Après avoir extrait les caractéristiques, la classification des sentiments est ensuite effectuée en utilisant plusieurs algorithmes de classification supervisés tels que machine à vecteurs de support (SVM), Naive Bayes (NB), Régression Logistic (RL), K-plus proches voisins (KNN), etc.
- Méthode basée sur le lexique : pour le modèle basé sur le lexique, ni les données étiquetées ni une étape d'apprentissage ne sont nécessaires pour concevoir le classifieur de sentiment. Le sentiment exprimé dans une phrase ou un document est déterminé à l'aide de lexiques de sentiments construits manuellement, prédéfinis ou traduits. Un lexique de sentiments contient des mots subjectifs avec leurs polarités (positives ou négatives) et leurs scores de polarité (Piryani *et al.*, 2017). Ainsi, la polarité d'un mot ou d'une phrase peut être décidée en utilisant son score sentimental dérivé du lexique.

Dans cette étude, on ne s'intéresse qu'aux méthodes à base d'apprentissage supervisé. Pour plus de détail sur l'impact des prétraitements combinés avec des méthodes basée sur le lexique sur l'AS du dialecte tunisien, vous pouvez vous référez à cette étude (Mulki *et al.*, 2018).

Considérant les travaux qui ont porté sur les dialectes arabes, peu de recherches ont porté sur le dialecte tunisien. Le dialecte arabe est généralement manipulé en utilisant des méthodes de traitement automatique de la langue (TAL) utilisées pour l'arabe standard moderne (ASM). Différentes techniques de prétraitement et différentes combinaisons de prétraitement ont été utilisés : la racinisation, la racinisation légère, l'élimination des mots vides et l'étiquetage d'émojis (Duwairi & El-Orfali, 2014; El-Beltagy *et al.*, 2017).

Quelques travaux ont été effectués sur l'AS en dialecte tunisien. Dans (Sayadi *et al.*, 2016), six classifieurs ont été entraîner avec différents types de n-grammes pour la classification de tweets issus

d'un corpus en arabe standard moderne et en dialecte tunisien. La meilleure performance pour la classification binaire à été obtenue avec l'algorithme SVM avec un F1-score de 63%.

Les auteurs dans (Medhaffar *et al.*, 2017b), ont employé les *documents embeddings* comme caractéristiques pour le modèle d'AS du dialecte tunisien. Les vecteurs obtenus sont utilisés pour entraînés des classifieurs SVM, Bernoulli NB (BNB) et perception multicouche (MLP). Les meilleurs résultats sont obtenus avec avec le classifieur MLP qui a atteint un F1-score de 78%.

3 Le modèle d'analyse de sentiment proposé

Dans cette étude, nous visons à déterminer parmi la racinisation, la racinisation légère, l'élimination des mots vides, l'utilisation des émojis et la prise en compte de la négation, le prétraitement ou la combinaison de prétraitements qui peuvent améliorer la performance de l'AS du dialecte tunisien. Par conséquent, nous pouvons décider avec quel(s) prétraitement(s) la reconnaissance des entités nommées doit être combinée de sorte que la performance de l'analyse des sentiments puisse être optimale.

L'analyse des sentiments des corpus tunisiens a été effectuée en utilisant l'outil Tw-StAR (Mulki *et al.*, 2017) qui se base sur un modèle d'apprentissage automatique supervisé au niveau des phrases. Trois variantes n-grammes de mots, y compris des unigrammes, des bigrammes et des trigrammes ont été adoptés comme caractéristiques pour entraîner les algorithmes de classification supervisés.

3.1 Prétraitement des données

Les étapes de prétraitement sont les suivantes :

- Prétraitement initial : Pour tous les corpus, une étape de prétraitement initiale commune qui inclut la suppression du contenu non porteur d'opinion tel que les URL, les noms d'utilisateurs, les dates, les chiffres, les symboles de hashtags et la ponctuation.
- Racinisation (Racine) : La racinisation est utilisée pour éliminer les suffixes et les préfixes des mots afin de gérer la variation morphologique des mots. Pour étudier l'effet des algorithmes de racinisation sur l'analyse de sentiment en tunisien, nous avons étudié la racinisation avec l'algorithme Farasa (Abdelali *et al.*, 2016) et la racinisation légère (Larkey *et al.*, 2002).
- Élimination des mots vides (Stop) : En raison de l'absence d'une liste de mots vide du dialecte tunisien, une liste de 1 661 mots vides de l'arabe standard moderne fournis par le groupe de TAL du Centre national de technologie informatique et de mathématiques appliquées de la cité du roi Abdulaziz pour la science et la technologie (KACST) [1] a été utilisée.
- Détection des émojis (Emoji) : Nous avons identifié deux types d'emoji les plus courants. Le premier type concerne les emojis positifs tels que le visage souriant, le visage avec larmes de joie, etc. Le deuxième type représente les emojis négatifs tels que le visage malheureux, le visage pensif, le visage inquiet, etc. Les emoji positifs sont remplacés par l'étiquette "PositiveEmoji" tandis que l'étiquette "NegtativeEmoji" est utilisée pour remplacer les emojis négatifs.
- Détection de la négation (Neg) : La négation est exprimée avec les indicateurs de mots arabes négatifs qui sont : " لا" (non), " لم" (n'ont pas), " لن" (ne sera pas), " لستُ" (je ne suis pas),

1. https ://github.com/abahanshal/arabic-stop-words-list1

" ليس" (ne pas), " دون" (sans)," لسن" (ne sont pas), " ليسوا" (ne sont pas), " بدون" (sans), " بلا" (sans), " أبدا" (jamais), " بغير" (sans), " غير" (n'est pas), " لستم" (vous n'êtes pas), " لستن" (vous n'êtes pas). Nous utilisons également des indicateurs de négation relatifs au dialecte tunisien : " ماكش" (tu n'es pas), " مانيش" (je ne suis pas), " ماكمش"(vous n'êtes pas)," مفماش" (il n'y a pas) and " ماهمش" (ils ne sont pas). Nous utilisons l'étiquette "NegWord" pour remplacer chaque négation.

3.2 Reconnaissance des entités nommées

Les entités nommées ont été traitées à l'aide du système de reconnaissance d'entités nommées fourni par (Gridach, 2016). Les entités nommées extraites ont ensuite été classées en positives ou négatives afin d'être marquées dans l'étape de prétraitement. Dans ce but, nous avons développé un algorithme d'assignation de polarité d'une entité nommée en fonction de ses informations contextuelles locales comme suit :

— Les entités nommées extraites des données d'apprentissage sont comparées avec les mots des phrases inclus dans les données d'apprentissage.
— Quand une correspondance entre une entité nommée spécifique et une phrase est trouvée, un score est attribué à cette entité nommée en fonction de la polarité de cette phrase telle que 1 est ajouté si la polarité de la phrase est positive tandis qu'un score de 1 est soustrait si la polarité de la phrase est négative.
— Ainsi, la polarité d'une entité nommée est déterminée par le signe du résultat de son score accumulé où les scores signés positifs et négatifs définissent les entités nommées positives et les entités nommées négatives respectivement.
— Quant aux entités nommées de scores nul, elles sont éliminées car elles sont également mentionnées dans les phrases positives et négatives.

3.3 Classification des sentiments

Le modèle d'AS supervisé est entraîné pour prédire la classe de polarité appropriée à des n-grammes d'entrée spécifiques. L'apprentissage est effectué avec l'algorithme Naive Bayes (NB) de scikit-learn [2] et l'algorithme Support Vector Machine (SVM) linéaire de LIBSVM [3].

4 Étude expérimentale

Dans les tableaux présentés, les performances obtenues pour les prétraitements simples ou combinés sont comparées aux résultats de référence qui représentent les performances obtenues par les systèmes de (Sayadi *et al.*, 2016), (Karmani, 2017) et (Medhaffar *et al.*, 2017a) que l'on note respectivement baseline 1, baseline 2 et baseline 3. Les macro mesures Précision, Rappel, F1-score et exactitude

2. http://scikit-learn.org/stable/modules/naive_bayes.html
3. https://www.csie.ntu.edu.tw/~cjlin/libsvm/

sont respectivement notés (P.), (R.), (F1.) et (Exa.). Nous avons utilisé 80% des données pour l'apprentissage et 20% pour le test.

Pour effectuer une comparaison objective avec les systèmes de références appliqués sur les corpus TEC, TAC et TSAC, nous avons dû utiliser les mêmes algorithmes de classification. Les algorithmes NB et SVM étaient utilisés pour les corpus TEC et TSAC tandis qu'un modèle à base de lexique a été utilisé pour TAC.

4.1 Corpus d'évaluation

Trois corpus avec un contenu collecté à partir des réseaux sociaux tunisiens ou mixtes tunisien-arabe standard moderne ont été utilisés :

— Corpus Électoral Tunisien (TCE) : ce corpus fait référence à un ensemble de 5 521 tweets collectés par (Sayadi *et al.*, 2016) lors des élections tunisiennes d'octobre 2014. Il combine arabe standard moderne et dialecte tunisien où les tweets tunisiens constituent la majorité des données. Après avoir réduit les tweets neutres, un jeu de données de 3 043 tweets est utilisé.
— Corpus d'analyse du sentiment Tunisien (TSAC) : un ensemble de données de 9 976 commentaires Facebook fournis par (Medhaffar *et al.*, 2017a). Ces commentaires représentent les réactions du public vis-à-vis des émissions de télévision tunisiennes populaires. Ils ont été annotés manuellement avec une polarité positive et négative. Dans cette étude, nous avons éliminé les instances Arabizi de cet ensemble de données de telle sorte que 7 366 commentaires sont utilisés.
— Corpus arabe tunisien (TAC) : Un ensemble de données composé de 800 tweets couvrant de multiples sujets tels que les médias, les télécommunications et la politique. Cet ensemble de données a été collecté par (Karmani, 2017) et annoté avec la polarité positive, négative et neutre. Nous n'avons traité que les cas positifs et négatifs de sorte que 746 tweets sont utilisés.

Nous n'avons pas fusionner les corpus d'évaluation puisque nous voulons examiner l'impact du prétraitement sur des corpus ayant un contenu tunisien ou sur un corpus ayant du contenu mixte MSA/tunisien.

L'élimination des tweets neutres des corpus TEC et TAC n'a pas empêché de faire une comparaison équitable puisque nous nous sommes comparé avec les résultats de classification binaires fournis par (Sayadi *et al.*, 2016) et avec les résultats d'évaluation de la classification binaire pour TAC (Karmani, 2017).

4.2 Résultats et discussion

Les techniques de prétraitement énumérées dans la section 2 ont été examinées une à une puis différentes combinaisons ont été appliquées. Cela a permis de définir la technique/combinaison de prétraitements qui a permis d'améliorer au mieux les performances de l'AS et donc de spécifier la technique/combinaison de prétraitements avec laquelle le marquage des entités nommées pourrait être intégré.

Trois variantes d'expériences ont été effectuées. La première consiste à utiliser toutes les caractéristiques n-grammes : unigrammes (uni), bigrammes (bi), trigrammes (tri) et leurs combinaisons (uni+bi, uni+ bi+tri), tandis que les deuxième et troisième expériences utilisent un nombre réduit des mêmes

caractéristiques résultant de l'utilisation de la fréquence avec deux valeurs de seuil égales respectivement à 2 et 3. Le tableau 1, le tableau 2 et le tableau 3 présentent les meilleures performances obtenues par les algorithmes NB ou SVM.

Prétraitement	**Caractéristiques**	**Algorithme**	**P.(%)**	**R.(%)**	**F1.(%)**	**Exa.(%)**
baseline 1	uni+bi	SVM	67	71	63	71.09
Stop	uni	SVM	72	70.5	70.6	71.6
Racine	uni	NB	75.3	73.4	73.6	74.5
Neg	uni+bi	SVM	75.7	71.7	71.7	73.4
Racine + Stop	uni	NB	75.7	73.3	73.4	74.5
Racine + ENs	uni	NB	**75.7**	**74**	**74.2**	**75**

TABLE 1 – Les performances du modèle supervisé pour le corpus TEC pour tous les prétraitements

Prétraitement	**Caractéristiques**	**Algorithme**	**P.(%)**	**R.(%)**	**F1.(%)**	**Exa.(%)**
baseline 2	morphologique	Lex	63	72.9	67.3	72.1
Stop	uni	NB	82.9	79.8	79.5	80
Racine	uni	SVM	86.3	85.9	85.9	86
Neg	uni+bi	SVM	86.6	85.9	85.9	86
Racine + Stop	uni+bi	NB	83.9	82.5	82.5	82.7
Neg + ENs	uni+bi	SVM	**87.4**	**86.6**	**86.6**	**86.7**

TABLE 2 – Les performances du modèle supervisé pour le corpus TAC pour tous les prétraitements.

Les résultats des tableaux 2 et 3 montrent clairement que SVM donne de meilleurs résultats que NB pour les corpus de moyenne et grande tailles tel que TAC et TSAC. Cela pourrait s'expliquer par la capacité de SVM à gérer la densité et la haute dimensionnalité des vecteurs de caractéristiques d'apprentissage. Cependant, le tableau 1 montre que les sentiments pour les corpus de petite taille (TEC) sont mieux classés par NB.

Il est à noter que l'utilisation de Farasa a permis d'améliorer la performance de la classification supervisée des sentiments pour les corpus TAC et TEC (tableau 1 et tableau 2) où le deuxième meilleur F1-score a été obtenu (85,9%) pour TAC avec une amélioration de 18,6% par rapport aux résultats de référence. Bien que Farasa a été entraîné avec des corpus d'arabe standard moderne, il a réussi à identifier les affixes à éliminer des mots tunisiens en raison du chevauchement lexical entre l'arabe standard moderne et les dialectes arabes en général (Samih *et al.*, 2017). Afin de conserver la variété des mots ayant la même racine et des significations différentes, nous avons également utilisé la racinisation légère. Néanmoins, çela n'a pas permis d'améliorer les performances pour tous les corpus, même lorsqu'il a été combiné avec d'autres techniques de prétraitement.

L'impact de l'élimination des mots vides sur l'analyse des sentiments est plus importante lorsque l'élimination des mots vides a été combinée avec la racinisation. Comme le montre le tableau 3, en utilisant le classificateur SVM sur TSAC, l'élimination des mots vides a conduit à une meilleure racinisation et donc à un deuxième meilleur F1-score égal à 93,8%. Comme le montre le tableau 1, pour le corpus TEC la précision est de 71,6% obtenue par l'élimination des mots vides uniquement et de 74,5% obtenue par la combinaison de la racinisation et l'élimination des mots vides.

Prétraitement	Caractéristiques	Algorithme	P.(%)	R.(%)	F1.(%)	Exa.(%)
baseline 3	doc embeddings	MLP	78	78	78	78
Stop	uni	SVM	92.5	92.3	92.4	92.6
Racine	uni	SVM	93.4	93.4	93.4	93.5
Neg	uni	SVM	92.6	92.5	92.5	92.7
Emo	uni	SVM	92.4	92.39	92.4	92.5
Racine + Stop	uni	SVM	93.8	93.8	93.8	93.9
Emo + Stop	uni	SVM	92.1	92.1	92.2	92.3
Emo + Racine	uni	SVM	**93.9**	**93.8**	**93.9**	**94**
Emo + Neg	uni	SVM	92.5	92.4	92.5	92.6
Emo + Racine + Stop	uni	SVM	93.8	93.8	93.8	93.9
Emo+ Racine + ENs	uni	SVM	92.8	92.86	92.8	93

TABLE 3 – Les performances du modèle supervisé pour le corpus TSAC pour tous les prétraitements.

La détection des emojis a été utilisée uniquement avec le corpus TSAC car les corpus TEC et TAC ne contiennent aucun emoji. Dans TSAC, l'étiquetage des emojis n'a pas eu un impact significatif sur la performance quand il était appliqué séparément alors que la combinaison avec la racinisation a obtenu le meilleur F1-score parmi toutes les expériences avec une valeur égale à 93,9%. De plus, la détection des emojis avec la négation a permis d'obtenir presque les mêmes résultats obtenus par la tâche de prétraitement de la négation. Cela pourrait être dû à un contenu sarcastique dans lequel les emojis n'expriment pas le vrai sens, mais son contraire.

Les tableaux 1, 2 et 3 montrent que les performances ont été améliorées pour tous les corpus lorsque la détection de négation a été appliquée. Néanmoins, la plus faible amélioration a été obtenue pour TEC, puisque l'exactitude a été améliorée de 2,31%, en comparaison aux améliorations de 13,9% et de 14,7% obtenues pour les corpus TAC et TSAC respectivement. Cela peut être expliqué par une meilleure précision dans la reconnaissance de la négation pour les corpus qui contiennent le tunisien seulement (TAC, TSAC) par rapport aux corpus aux contenus mixtes tunisien/arabe standard moderne tel que TEC.

L'étiquetage des entités nommées combiné avec la négation pour le corpus TAC et avec la racinsation pour le corpus TEC ont amélioré le F1-score de 6,7% et 4,8% pour les corpus TAC et TEC respectivement.

5 Conclusion

Cet article a mis en évidence le rôle essentiel de la phase de prétraitement dans l'analyse des sentiments du dialecte tunisien. L'évaluation de diverses techniques de prétraitement a démontré qu'en présence d'émoji, la racinisation et l'étiquetage des emojis est la meilleure combinaison. Ainsi, combiner la technique d'étiquetage des entités nommées avec les techniques les plus efficaces a conduit aux meilleures performances d'AS de telle façon que les résultats de références ont été dépassés par une marge significative. Pour les travaux futurs, les performances de l'AS peuvent être encore améliorées si la stratégie de détection de la négation était étendue pour traiter l'ironie et le contenu sarcastique.

Références

ABDELALI A., DARWISH K., DURRANI N. & MUBARAK H. (2016). Farasa : A fast and furious segmenter for arabic. In *Proceedings of the Demonstrations Session, NAACL HLT 2016, The 2016 Conference of the North American Chapter of the Association for Computational Linguistics : Human Language Technologies, San Diego California, USA, June 12-17, 2016*, p. 11–16.

AKAICHI J. (2014). Sentiment classification at the time of the tunisian uprising : Machine learning techniques applied to a new corpus for arabic language. In *Proceedings of the 2014 European Network Intelligence Conference*, ENIC '14, p. 38–45.

DUWAIRI R. M. & EL-ORFALI M. (2014). A study of the effects of preprocessing strategies on sentiment analysis for arabic text. *J. Information Science*, **40**(4), 501–513.

EL-BELTAGY S. R., KALAMAWY M. E. & SOLIMAN A. B. (2017). Niletmrg at semeval-2017 task 4 : Arabic sentiment analysis. In *Proceedings of the 11th International Workshop on Semantic Evaluation, SemEval@ACL 2017, Vancouver, Canada, August 3-4, 2017*, p. 790–795.

GRIDACH M. (2016). Character-aware neural networks for arabic named entity recognition for social media. In *Proceedings of the 6th Workshop on South and Southeast Asian Natural Language Processing (WSSANLP2016)*, p. 23–32 : The COLING 2016 Organizing Committee.

KARMANI N. (2017). *Tunisian Arabic Customer's Reviews Processing And Analysis For an Internet Supervision System*. PhD thesis, Sfax University, Tunisia.

LARKEY L. S., BALLESTEROS L. & CONNELL M. E. (2002). Improving stemming for arabic information retrieval : light stemming and co-occurrence analysis. In *SIGIR 2002 : Proceedings of the 25th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, August 11-15, 2002, Tampere, Finland*, p. 275–282.

MEDHAFFAR S., BOUGARES F., ESTÈVE Y. & HADRICH-BELGUITH L. (2017a). Sentiment analysis of tunisian dialects : Linguistic ressources and experiments. In *Proceedings of the Third Arabic Natural Language Processing Workshop*, p. 55–61, Valencia, Spain : Association for Computational Linguistics.

MEDHAFFAR S., BOUGARES F., ESTÈVE Y. & HADRICH-BELGUITH L. (2017b). Sentiment analysis of tunisian dialects : Linguistic ressources and experiments. In *Proceedings of the Third Arabic Natural Language Processing Workshop*, p. 55–61, Valencia, Spain : Association for Computational Linguistics.

MULKI H., HADDAD H., ALI C. B. & İSMAIL BABAOĞLU (2018). Tunisian dialect sentiment analysis : A natural language processing-based approach. *Computación y Sistemas. ISSN14055546*.

MULKI H., HADDAD H., GRIDACH M. & BABAOGLU I. (2017). Tw-star at semeval-2017 task 4 : Sentiment classification of arabic tweets. In *Proceedings of the 11th International Workshop on Semantic Evaluation, SemEval@ACL 2017, Vancouver, Canada, August 3-4, 2017*, p. 664–669.

PIRYANI R., DEVARAJ M. & SINGH V. K. (2017). Analytical mapping of opinion mining and sentiment analysis research during 2000-2015. *Inf. Process. Manage.*, **53**(1), 122–150.

SAMIH Y., ATTIA M., ELDESOUKI M., ABDELALI A., MUBARAK H., KALLMEYER L. & DARWISH K. (2017). A neural architecture for dialectal arabic segmentation. In *Proceedings of the Third Arabic Natural Language Processing Workshop, WANLP 2017@EACL, Valencia, Spain, April 3, 2017*, p. 46–54.

SAYADI K., LIWICKI M., INGOLD R. & BUI M. (2016). Tunisian dialect and modern standard arabic dataset for sentiment analysis : Tunisian election context. In *Second International Conference on Arabic Computational Linguistics, ACLING 2016, Konya, Turkey, 7-8 April 2016*, p. 35–53.

YASAVUR U., TRAVIESO J., LISETTI C. L. & RISHE N. D. (2014). Sentiment analysis using dependency trees and named-entities. In *Proceedings of the Twenty-Seventh International Florida Artificial Intelligence Research Society Conference, FLAIRS 2014, Pensacola Beach, Florida, May 21-23, 2014.*

Detecting context-dependent sentences in parallel corpora

Rachel Bawden[1] Thomas Lavergne[1] Sophie Rosset[2]
(1) LIMSI, CNRS, Univ. Paris-Sud, Université Paris-Saclay, F-91405 Orsay, France
(2) LIMSI, CNRS, Université Paris-Saclay, F-91405 Orsay, France
lastname@limsi.fr

RÉSUMÉ

Détection dans des corpus parallèles de phrases dépendantes du contexte

Dans cet article, nous proposons plusieurs approches pour l'identification automatique de phrases parallèles qui nécessitent du contexte linguistique extra-phrastique pour être correctement traduites. Notre objectif à long terme est de construire de façon automatique un jeu de test de phrases dépendantes du contexte afin d'évaluer les modèles de traduction automatique conçus pour améliorer la traduction de phénomènes discursifs et contextuels. Nous fournissons une discussion et une critique qui montrent que les approches actuelles ne nous permettent pas d'atteindre notre but et qui suggère que l'évaluation individuelle de phenomènes est probablement la meilleure solution.

ABSTRACT

In this article, we provide several approaches to the automatic identification of parallel sentences that require sentence-external linguistic context to be correctly translated. Our long-term goal is to automatically construct a test set of context-dependent sentences in order to evaluate machine translation models designed to improve the translation of contextual, discursive phenomena. We provide a discussion and critique that show that current approaches do not allow us to achieve our goal, and suggest that for now evaluating individual phenomena is likely the best solution.

MOTS-CLÉS : traduction automatique, contexte, evaluation, discours.

KEYWORDS: machine translation, context, evaluation, discourse.

1 Introduction

Recent work in Machine Translation (MT) has focused on using information beyond the current sentence boundary to aid translation (Libovický & Helcl, 2017; Wang *et al.*, 2017; Jean *et al.*, 2017). The aim of these *contextual* MT systems is to remedy the flaw of traditional MT of translating sentences independently of each other, in particular to improve the translation of discourse phenomena. Despite the progress made in incorporating linguistic context into MT (Bawden *et al.*, 2018), these gains are often not observable using automatic evaluation metrics, such as BLEU (Papineni *et al.*, 2002), and manual analysis of translations is often anecdotal. Whilst strategies such as producing contrastive sentence pairs to be reranked by MT models is a promising strategy for evaluation (Rios Gonzales *et al.*, 2017; Bawden *et al.*, 2018), producing the test sets is often time-consuming and unrepresentative of real data. Morevoer, the distinction is often lacking between examples that need extra-sentential context to be translated and those that do not.

A very useful addition to the test suites available would therefore be a test set of real, attested

examples that require extra-sentential linguistic context to be correctly translated, as this would enable us to evaluate the progress made by contextual MT models specifically on the most difficult examples. Since manually identifying real sentences is very time-consuming, our long-term goal is to automatically construct such a test set. In this paper, we aim to show that designing and implementing a method of automatically detecting these sentences in a parallel corpus remains problematic, as shown by a reflection on what such a method would entail and preliminary experiments using tools currently at our disposal to implement it.

We begin by discussing the types of phenomena we wish to identify (Section 2) and existing work on evaluating discourse phenomena (Section 3). We then define the goals and principles such an identification method would adhere to (Section 4). Finally, in Section 5, we critique two possible approaches to the problem, suggesting theoretical limitations of each approach. Our hope is for this work to provide the basis for discussion on modelling contextual phenomena in a multilingual setting, in a view to automatically identifying context-dependent sentences in the long term.

2 Context-dependent phenomena

In practice, many sentences can be correctly translated in isolation, without surrounding context, which explains why most MT systems today translate sentences independently of each other. However certain phenomena, mostly related to discourse, whose scope, by definition is defined at the discourse rather than the sentence level of discourse, cannot be systematically and correctly translated without extra context. Examples include anaphoric pronoun translation (Hardmeier & Federico, 2010; Guillou, 2016; Loaiciga Sanchez, 2017), lexical disambiguation (Carpuat & Wu, 2007; Rios Gonzales *et al.*, 2017) and cross-lingual discourse connective prediction (Meyer & Popescu-Belis, 2012). These phenomena have a common characteristic: they are cross-lingually ambiguous and can only be disambiguated with the help of linguistic context. This linguistic context can appear within the sentence containing the ambiguous element or elsewhere in the text, in which case we refer to it as extra-sentential linguistic context.

Cross-lingual ambiguity occurs because of mismatches in the language systems of the source and target languages, such that there are several translations possible out of context and only one correct in context.[1] The ambiguity can be morphological, syntactic, semantic and/or discursive. One example of the morphological level is the translation of anaphoric pronouns, which poses difficulties in MT due to structural differences in gender marking cross-lingually. For example, the French translation of English *it* is ambiguous between variants *il* (masc.) and *elle* (fem.), depending on the gender of the French noun with which the pronoun corefers. On the syntactic level, ambiguity can arise from an inherent ambiguity in the source language that is not preserved in the target language. For example, English 'green chestnuts and pears' is ambiguous between French 'des marrons verts et des poires' (only the chestnuts are green) and 'des marrons et des poires verts' (chestnuts and pears are green). Cross-lingual semantic ambiguity is where semantic ambiguity in the source language is not preserved in the target language and a choice must be made between the different meanings. For example, the English word *spade* is ambiguous between French *bêche* 'gardening implement' and *pique* 'suit of cards' (Cf. work on word sense disambiguation in MT by Carpuat & Wu (2007)).[2]

[1] This is different from the choice between synonyms or paraphrases, where any of the choices may be a correct translation.

[2] Although this example may seem contrived, in reality, in spoken dialogue scenarios, where utterances can be very short (E.g. "You've got a spade?"), more ambiguity can be expected.

Finally, at the discursive level, elements such as discourse connectives are often language-specific and are often expressed differently cross-lingually (Cf. work on implicitation of discourse connectives by Meyer & Webber (2013)).

Although the examples given above are relatively well-studied phenomena, particularly in a monolingual setting (coreference resolution, word sense disambiguation, discourse relation labelling, etc.), this cannot be seen as an exhaustive list of context-dependent phenomena. Ideally, it would be useful to study translation quality on all types of context-dependent sentences, not just those in a pre-defined list, especially as they are dependent on the particular language pair.

3 Evaluating discourse in MT

Evaluating discourse and other context-dependent phenomena in MT poses a problem for two main reasons.[3] Firstly, most sentences do not require context to be translated. When they do, few words are affected by an incorrect translation relative to the total number of words in the dataset, despite the fact that these errors can be seriously detrimental to the understanding of the translation.[4] Secondly, the correct translation of certain discourse phenomena, including anaphoric pronoun prediction, depends on previously made translation choices (ensuring translation coherence), ruling out metrics that rely on comparing surface forms of the predicted translation with a reference translation.

As interest in contextual MT surges, the question of how to correctly evaluate the impact of the added context has not been far behind, with different solutions for evaluation, both manual and automatic, being proposed, aimed to overcome the problems described above. In terms of manual evaluation, the aim has been to construct a corpus containing only examples of interest in order to combat the sparsity problem cited above. Isabelle *et al.* (2017) provides such a set of test examples designed to test different well-known problems faced by MT systems, including discourse phenomena. Two solutions have been proposed for automatic evaluation of specific phenomenon. The first, adopted by shared task organisers for both the cross-lingual pronoun prediction task at WMT'16 and DiscoMT'17 (Guillou *et al.*, 2016; Loáiciga *et al.*, 2017) and the cross-lingual word sense disambiguation (WSD) task at SemEval-2013 (Lefever & Hoste, 2013), is to change the nature of the task, and to evaluate the models' ability to translate solely the word of interest, whilst the rest of the translation is imposed for all contestants. This alleviates the second problem of translation coherence. The second automatic method, which also involves avoiding comparison of different models' translations, is to evaluate the capacity of MT models to rerank different hypotheses. Presenting models with contrastive pairs of examples and comparing the models on their ability to rank the correct hypothesis higher than an incorrect one is a way of indirectly evaluating them (Cf. (Sennrich, 2017) for grammatical errors (Rios Gonzales *et al.*, 2017) for WSD and (Bawden *et al.*, 2018) for coreference and lexical coherence/cohesion.

Aside from (Bawden *et al.*, 2018), which imposes that the disambiguating context occur in the previous sentence, the other automatic evaluation methods do not control for the fact that the disambiguating context can appear within the current sentence or beyond the sentence boundary.[5] This means that many examples in the sets can be resolved using sentence-internal context and therefore do

[3] Cf. (Hardmeier, 2012) for a detailed overview of problems faced.

[4] E.g. a coreference error typically leads to one mistranslated pronoun, which changes the entire meaning of the sentence.

[5] The number of pronouns with intra-sentential antecedents was roughly equal to the number with extra-sentential antecedents in the DiscoMT2015 test set (Guillou, 2016, pp. 161).

not directly evaluate the ability of contextual models to use context beyond the current sentence. This notably proved problematic for the evaluation of the 2016 pronoun task, of which the highest performing model did not use any extra-sentential context (achieved higher scores based on the inter-sentential examples alone). A useful complement to these test suites would therefore be a method of automatically constructing a test set of sentences that require linguistic context to be correctly translated. The advantages of such a method would be its automatic nature, given the difficulty of manually finding representative examples of context-dependent phenomena and the fact that it could potentially find more diverse phenomena than a human annotator is capable of finding.

4 Automatic context-dependent sentence detection

Our long-term goal is to propose and develop a method of identifying real corpus examples that are cross-lingually ambiguous and necessarily require extra-sentential context (as opposed to intra-sentential context) to be correctly translated.[6] In theory, such a method would separate parallel sentences for which all information needed to produce the target sentence is found within the source sentence (non-context-dependent) from those for which part of the information can only be found in the surrounding sentences (context-dependent).

4.1 Goals and principles

To achieve our goal, the ideal method would adhere to a certain number of principles to ensure (i) the unbiased nature of the test set, (ii) diversity and a large coverage of the phenomena detected and (iii) easy transferability to other language pairs. Although these properties may not be mutually attainable, attempting to adhere to these three properties is key to developing a detection method.

(i) Unbiased test set A test set should be inherently unbiased towards a certain MT model or a certain type of model if it is to be used to fairly evaluate and compare models. This means that, ideally, the detection method itself should not rely on an existing MT model whose goal is to accomplish a task that the test set is designed to test. In our specific case, this means that any use of contextual MT models would violate this principle.

(ii) Diversity and large coverage of phenomena A number of cases have been previously identifed in the literature as requiring context to be correctly translated, for example anaphoric pronouns, lexical ambiguity, discourse connectives, other cases of lexical cohesion. However, in practice, the main focus has been on only a couple of these phenomena, namely anaphoric pronoun resolution, and to a lesser extent lexical ambiguity. It is therefore interesting to keep the method as generic as possible, giving us the opportunity of identifying new context-dependent phenomena.

(iii) Easy transferability to other language pairs To ensure that similar test sets can be easily produced for other language pairs, the detection method should be independent or at least only weakly dependent on the language pair. Since the majority of contextual phenomena depend on the language systems of the source and target language, this third point complements the previous point concerning the diversity of linguistic phenomena; the less *a priori* knowledge of the language pair required, the more adaptable the method will be to new language pairs, for which we do not have such knowledge.

[6] We make the approximation that we can judge whether a source sentence is cross-lingually ambiguous based on the reference translation in the parallel corpus. In reality a translation can be chosen that avoids the ambiguity altogether.

The question is, is such a method currently possible?

5 Comparison of methods

An ideal method would be one relying on complete and comparable representations of the source sentence and of the target sentence both with and without linguistic context. Intuitively, for context-dependent sentences to be correctly translated, the information present in the representation of the target sentence would be impossible to reconstruct from the representation of the source sentence, unless the information from the context is also included. We look at two different approaches for simulating this idealised scenario, working (i) at the sentence level and (ii) at the word level.

Modelling at the sentence level

Following promising work on distributional representations of words (Mikolov *et al.*, 2013; Pennington *et al.*, 2014), recent work has emerged on the distributional representation of larger units of text, such as sentences. These representations are meant to encode generic, often semantic information about the sentence in fixed-size vectors. If sentence embeddings can encode information about a sentence, can they provide the necessary framework to determine whether or not a target sentence is translatable from its source sentence alone, ignoring its context? A positive answer to this question would require the following to be true: (i) a neural network model can be trained to predict the target sentence embedding from the source sentence embedding; a poor prediction for a given source embedding would be a sign that all the information necessary to produce its corresponding target embedding is not present in the source embedding; (ii) a second model trained to predict the target sentence embedding from a joint embedding of the source sentence and its context (source- or target-side) would predict a better target embedding for this context-dependent sentence.

The problem with this method is the number of assumptions that are made: (i) the sentence embedding fully represents the sentence, (ii) a mapping can be learnt between source and target sentence embeddings, (iii) we have a reliable metric to evaluate whether the contextually predicted sentence embedding is significantly more similar to the real target sentence embedding than the non-contextual one. Preliminary exploratory experiments in this direction which aimed to learn the mapping between DOC2VEC embeddings (Mikolov *et al.*, 2013) in the source and target languages using a small feedforward neural network confirmed that these assumptions were too great. One fundamental flaw with such an approach is that we have little control over the type of information stored in the representation, and no guarantee that this information will be useful for predicting cross-lingual ambiguity. With no control over the type of information modelled, evaluating whether the predicted embedding is sufficiently similar to the true target representation is also an open problem, and makes the method untractable. Given an imperfect representation of a sentence, judging whether a prediction is more similar to the target representation than another is impossible without knowing on what criteria we base the similarity. The approach could only really work with a near-perfect representation of all the information in a sentence, or more control over what kind of information is stored. Given our very generic aim to identify all types of context-dependent phenomena, this approach is not yet feasible. The problem is almost circular; if we had a method to perfectly map the representation of a sentence in context from one language to another, machine translation itself would be a solved task.

Modelling at the word level

Given the problem of obtaining sufficiently complete sentence-level embedding representations, a

reasonable compromise is to try to work at the word level. We therefore consider a second, reduced approach, this time assuming that the ambiguity arises from a single word in the source sentence and only affects its translation in the target sentence.[7] We therefore also need to make the assumption that we have a method to identify sentences containing an ambiguous element. This splits the problem into two steps: (i) identifying sentences containing ambiguous elements, and (ii) separating the sentences which do not need extra context to be translated from those that do. Given that methods exist to detect specific phenomena in corpora, e.g. anaphoric pronouns (Hardmeier *et al.*, 2015) and semantically ambiguous words (Rios Gonzales *et al.*, 2017), we suppose that new methods can be developed for more phenomena. This reduces the task to identifying whether the disambiguating context is found within the sentence, in the neighbouring sentences or cannot be found in the text at all.

An approach at the word level would typically look at the probability of the ambiguous target word given just the current sentence and also given sentence-external context, compared to the probability of the alternative, incorrect solution(s). For example, in the *le chat miaule et [ ] dort* for "the cat meow and **it** sleeps", where the target word is *il* and the incorrect alternative *elle*, we would expect the target word to have a higher probability than the alternative word, regardless of the addition of extra context, since coreference is resolved within the sentence. In a sentence where the antecedent appears in the previous sentence, we would expect the probability of the target word to increase with the addition of this previous sentence relative to the probability of the alternative translation.

Yet again, this method suffers from strong assumptions about the capacity of current NLP models to use contextual information to make such predictions. The assumptions were confirmed to be false by exploratory experiments using tool CONTEXT2VEC (Melamud *et al.*, 2016), which can be used like a language model to make predictions about the form of a target word given a certain context. Three limitations were observed: (i) the intrinsic probability of a word, as determined by its frequency, has a very large effect on its probability in context, making it very complicated to assess the effect of adding context, (ii) the capacity of generic language models to model complex and structured problems such as coreference chains is insufficient, even for simple, short utterances, and (iii) in light of the second limitation, all context, even if not directly relevant to the translation of the ambiguous word, has an effect on the probability of the word. We have little control over which information is considered important by the model, particularly if we wish to keep the approach as general as possible.

6 Conclusion

We have described and motivated a theoretically interesting task of identifying sentences that are cross-lingually ambiguous and dependent on extra-sentential linguistic context. Beyond translation, this could have a wider impact on NLP applications, including dialogue generation and understanding. Through a reflection on the pre-requisites for such a detection method, and by exploring two different approaches to the problem, we have found that the task is very ambitious. The limitations identified have shown us that as long as complete and robust representations of all information within sentences are not yet achievable, the task of identifying context-dependent sentences using a method that is agnostic to the type of phenomenon is unlikely to be attainable. For now it appears that detecting contextual phenomena is better performed on a per-phenomenon basis.

[7] This assumption works at least for most of the cases cited in Section 2.

References

BAWDEN R., SENNRICH R., BIRCH A. & HADDOW B. (2018). Evaluating Discourse Phenomena in Neural Machine Translation. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics; Human Language Technologies*, New Orleans, Louisiana, USA. To appear.

CARPUAT M. & WU D. (2007). Context-Dependent Phrasal Translation Lexicons for Statistical Machine Translation. In *Proceedings of the 11th Machine Translation Summit*, p. 73–80, Copenhagen, Denmark.

GUILLOU L. (2016). *Incorporating Pronoun Function into Statistical Machine Translation*. PhD thesis, School of Informatics. University of Edinburgh.

GUILLOU L., HARDMEIER C., NAKOV P., STYMNE S., TIEDEMANN J., VERSLEY Y., CETTOLO M., WEBBER B. & POPESCU-BELIS A. (2016). Findings of the 2016 WMT Shared Task on Cross-lingual Pronoun Prediction. In *Proceedings of the 1st Conference on Machine Translation*, WMT'16, p. 525–542, Berlin, Germany.

HARDMEIER C. (2012). Discourse in statistical machine translation. a survey and a case study. *Discours [online]*, **11**.

HARDMEIER C. & FEDERICO M. (2010). Modelling pronominal anaphora in statistical machine translation. In *Proceedings of the 7th International Workshop on Spoken Language Translation*, IWSLT'10, p. 283–289, Paris, France.

HARDMEIER C., NAKOV P., STYMNE S., TIEDEMANN J., VERSLEY Y. & CETTOLO M. (2015). Pronoun-Focused MT and Cross-Lingual Pronoun Prediction: Findings of the 2015 DiscoMT Shared Task on Pronoun Translation. In *Proceedings of the 2nd Workshop on Discourse in Machine Translation*, DISCOMT'15, p. 1–16, Lisbon, Portugal.

ISABELLE P., CHERRY C. & FOSTER G. (2017). A Challenge Set Approach to Evaluating Machine Translation. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, EMNLP'17, p. 2476–2486, Copenhagen, Denmark.

JEAN S., LAULY S., FIRAT O. & CHO K. (2017). Does Neural Machine Translation Benefit from Larger Context? In *arXiv:1704.05135*. arXiv: 1704.05135.

LEFEVER E. & HOSTE V. (2013). SemEval-2013 Task 10: Cross-lingual Word Sense Disambiguation. In *Proceedings of the 2nd Joint Conference on Lexical and Computational Semantics*, p. 158–166, Atlanta, Georgia.

LIBOVICKÝ J. & HELCL J. (2017). Attention Strategies for Multi-Source Sequence-to-Sequence Learning. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics*, ACL'17, p. 196–202, Vancouver, Canada.

LOAICIGA SANCHEZ S. (2017). *Pronominal anaphora and verbal tenses in machine translation*. PhD thesis, University of Geneva.

LOÁICIGA S., STYMNE S., NAKOV P., HARDMEIER C., TIEDEMANN J., CETTOLO M. & VERSLEY Y. (2017). Findings of the 2017 DiscoMT Shared Task on Cross-lingual Pronoun Prediction. In *Proceedings of the 3rd Workshop on Discourse in Machine Translation*, DISCOMT'17, p. 1–16, Copenhagen, Denmark.

MELAMUD O., GOLDBERGER J. & DAGAN I. (2016). context2vec: Learning Generic Context Embedding with Bidirectional LSTM. In *Proceedings of the 20th SIGNLL Conference on Computational Natural Language Learning*, CoNLL'16, p. 51–61, Berlin, Germany.

MEYER T. & POPESCU-BELIS A. (2012). Machine Translation of Labeled Discourse Connectives. In *Proceedings of the 10th Biennial Conference of the Association for Machine Translation in the Americas*, AMTA'12, p. 129–138, San Diego, California, USA.

MEYER T. & WEBBER B. (2013). Implicitation of Discourse Connectives in (Machine) Translation. In *Proceedings of the 1st Workshop on Discourse in Machine Translation*, DISCOMT'13, p. 19–26, Sofia, Bulgaria.

MIKOLOV T., SUTSKEVER I., CHEN K., CORRADO G. & DEAN J. (2013). Distributed representations of words and phrases and their compositionality. In *Proceedings of the 26th Annual Conference on Neural Information Processing Systems*, NIPS'13, p. 3111–3119, Lake Tahoe, USA.

PAPINENI K., ROUKOS S., WARD T. & ZHU W.-J. (2002). BLEU: A Method for Automatic Evaluation of Machine Translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, ACL'02, p. 311–318, Philadelphia, Pennsylvania, USA.

PENNINGTON J., SOCHER R. & MANNING C. D. (2014). GloVe: Global Vectors for Word Representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing*, EMNLP'14, p. 1532–1543, Doha, Qatar.

RIOS GONZALES A., MASCARELL L. & SENNRICH R. (2017). Improving Word Sense Disambiguation in Neural Machine Translation with Sense Embeddings. In *Proceedings of the 2nd Conference on Machine Translation*, WMT'17, p. 11–19, Copenhagen, Denmark.

SENNRICH R. (2017). How Grammatical is Character- level Neural Machine Translation? In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics*, EACL'17, p. 376–382, Valencia, Spain.

WANG L., TU Z., WAY A. & QUN LIU (2017). Exploiting Cross-Sentence Context for Neural Machine Translation. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, EMNLP'17, p. 2816–2821, Copenhagen, Denmark.

Prédiction de l'échec d'une conversation médiée dans un contexte de dialogues à rôles asymétriques

R. Carbou[1a, 2] D. Charlet[1b] G. Damnati[1b] F. Landragin[2] J.-L. Bouraoui[1b]
(1) Orange Labs, Châtillon (a), Lannion (b), France
(2) Lattice, CNRS, ENS Paris, Université Sorbonne Nouvelle, PSL Research University, USPC, 1 rue Maurice Arnoux, 92120 Montrouge
romain.carbou@orange.com, delphine.charlet@orange.com, geraldine.damnati@orange.com, frederic.landragin@ens.fr, jeanleon.bouraoui@orange.com

RESUME

Dans une conversation humain-humain entre un usager et un interlocuteur en centre d'assistance, on se place dans le contexte où l'issue du dialogue est caractérisée par une notion de succès ou d'échec, explicitement annotée ou extrapolée. L'étude envisage différents paramètres susceptibles d'exercer une influence sur un modèle de classification prédictive des échecs constatés. On cherchera d'une part à exploiter une modélisation de la distribution lexicale tirant parti de l'asymétrie des rôles des locuteurs. On examinera d'autre part si la partie du lexique plus étroitement liée au domaine d'assistance client abordé ici, modifie la qualité de la prédiction. On interrogera enfin les perspectives de généralisation du modèle à des corpus morphologiquement comparables.

ABSTRACT

In a human-to-human conversation between a user and his interlocutor in an assistance center, we suppose a context where the conclusion of the dialog can characterize a notion of success or failure, explicitly annotated or deduced. The study involves different approaches expected to have an influence on predictive classification model of failures. On the one hand, we will aim at taking into account the asymmetry of the speakers' roles in the modelling of the lexical distribution. On the other hand, we will determine whether the part of the lexicon most closely relating to the domain of customer assistance studied here, modifies the quality of the prediction. We will eventually assess the perspectives of generalization to morphologically comparable corpora.

MOTS-CLES : dialogue humain-humain, échec d'un dialogue, corpus de dialogues, apprentissage artificiel, évaluation de dialogues, dialogue asymétrique.

KEYWORDS: human-to-human dialog, dialog failure, dialog corpus, artificial learning, dialogue evaluation, asymmetric dialog.

1 Introduction

Dans un corpus de dialogues humain-humain à rôles *asymétriques* (e.g. un utilisateur et un agent au sein d'un service client), on cherche à prédire les issues d'« échec » au regard d'une certaine définition. Pour cela on utilisera, de façon nominale, une méthode de classification supervisée utilisant les annotations de sortie disponibles, et dont on répétera les phases d'entraînement et de test en faisant varier différents paramètres. Chaque donnée d'observation est un dialogue, représenté

sous la forme d'un sac de mots dont on compte le nombre d'occurrences au sein de ce dialogue. L'objectif poursuivi est de pouvoir prédire l'échec de futurs dialogues se déroulant dans le même environnement mais aussi d'examiner si des phénomènes stables permettent d'utiliser le modèle sur des corpus de dialogues différents, selon une approche guidée par corpus [Tognini-Bonelli, 2001]. Le corpus étudié, *Datcha* [Damnati & al, 2016], rassemble des dialogues écrits (« tchats ») entre clients et téléconseillers du service technique de la société Orange.

On exploitera en particulier l'asymétrie des rôles du client (C) et du téléconseiller (T) sous l'angle lexical. Dans un cas, on considèrera dans sa globalité la distribution des mots employés, quel que soit le locuteur. Dans l'autre cas, les distributions propres à C ou à T seront considérées séparément. On parlera alors de données d'observations « différenciées » (par locuteur).

En second lieu, la taille du lexique obtenu directement par segmentation des mots est supérieure à 11000 flexions. Il s'agira d'en choisir des critères de réduction pertinents – pour, d'une part, garder un lexique significatif et, d'autre part, assurer un bon comportement de l'algorithme d'apprentissage du modèle. Or, l'univers dialogique constitue ici un « domaine fermé » (l'assistance client d'un opérateur), en opposition au « domaine ouvert » (lequel met en jeu des dialogues portant sur des sujets d'une diversité arbitraire). On examinera l'influence relative de la part du lexique la plus caractéristique du domaine (ci-après désignée comme part « thématique »), en retranchant optionnellement du lexique un sous-ensemble de 815 mots usuels (articles, mots de liaison syntaxique, et mots appartenant au vocabulaire général). Les données d'observation ainsi obtenues seront ci-après qualifiées de « filtrées ».

La segmentation du texte en mots est réalisée par un outil de lemmatisation (voir § Méthode). Celui-ci permettra, en troisième lieu, de comparer la forme originale et la forme lemmatisée de la distribution lexicale. Cette dernière variante a un fondement purement syntaxique, agnostique vis-à-vis de l'asymétrie exploitée par les deux précédentes. On notera que le lemmatiseur reconnaît directement certaines expressions locutionnelles, comme par exemple « prendre en charge » (comptant alors pour une entrée dans le lexique). Enfin, on ne corrigera pas les déviations linguistiques propres à Datcha, telles que discutées dans [Damnati & al, 2016].

L'action individuelle de ces trois variantes de construction de la distribution lexicale sera comparée à la distribution nominale *non différenciée par locuteur*, *non filtrée*, et *non lemmatisée*. Un enjeu de généralisation sera de rechercher les conditions où le contraste sur les résultats est le plus marqué, à savoir dans une fenêtre où l'ensemble du lexique n'est construit ni trop finement (« surabondance » lexicale) ni trop grossièrement (« pénurie » lexicale).

La section 2 présente le corpus Datcha, ses caractéristiques globales et ses annotations. Les sections 3 et 4 décrivent respectivement la méthodologie d'apprentissage du modèle prédictif et les résultats obtenus dans chacune des variantes. La section 5 commente et interprète ces résultats. Enfin, la section 6 propose de futures extensions à l'étude menée.

2 Corpus

La problématique d'annotation d'un corpus de dialogue à des fins d'évaluation s'étend sur une longue période, de MADCOW [Hirschman, 1992] à ADEM [Lowe & al, 2017]. Ici, le corpus est constitué de 2775 dialogues de type tchat, issus d'un centre de contact en Assistance Technique.

Les dialogues, lors d'études précédentes sur ce même corpus, ont été annotés manuellement par un expert. Celui-ci a annoté chaque dialogue selon le type de résolution du problème client tel qu'il est observé à l'issue de la conversation :

- *échange* : T propose un remplacement matériel ;
- *résolu* : T a fourni l'information attendue ou corrigé le problème ;
- *à tester* : T fournit un scénario de résolution en cours de session, avec actions ultérieures ;
- *hors périmètre* : le problème de C n'est pas du ressort de T, dans ce cas une réorientation est le plus souvent proposée, mais il peut ne pas y avoir réorientation selon circonstances ;
- *pas de solution* : la conversation n'a abouti à aucune des résolutions précédentes.

Le succès ici n'est pas lié à l'obtention au cours du dialogue d'un ensemble d'informations caractéristiques (« slot-filling ») comme abordé par [Claveau & al, 2013] ou [Talha & al, 2014]. Le téléconseiller a pour tâche d'établir un diagnostic qui par nature induit une infinité de possibles. Aussi, on considère que toute annotation exprimant un progrès dans la résolution de la demande initiale est un succès. Ainsi, un dialogue sera considéré comme échec s'il est annoté en « *pas de solution* » ou « *hors-périmètre sans réorientation* ». Ces situations sont marginales (une dizaine d'occurrences) et la majorité des conversations considérées en échec sont celles qui n'aboutissent pas. Les raisons sont diverses : le dialogue s'est interrompu inopinément (78% des cas) ; le client raccroche de son plein gré (12% des cas) ou devient définitivement inactif (8% des cas). La tâche étudiée dans cet article est donc la prédiction de l'échec, avec une répartition des conversations en 76,6% de succès et 23,4% d'échecs.

Dans ce contexte où la variable à prédire n'a pas été directement annotée, on voit que la tâche définitoire trahit des difficultés intrinsèques propres à la détermination de la « coopération » entre les acteurs [Grice, 1975].

Remarque : C'est l'échec qui est ici pris comme modalité positive. Or, les phénomènes qui orientent vers le succès sont aussi légitimes à étudier. En fait, ceux-ci semblent se concentrer sur des aspects cérémoniels (marque de politesse, satisfaction, etc.). Par exemple le test du seul mot « merci » produit pour la modalité négative, *c'est-à-dire le succès*, un triplet (rappel ; précision ; F-mesure) de (82,41% ; 94,88% ; 88,20%). Donc dès lors que le succès devient l'objet de la prédiction, *un seul mot* fait mieux que le modèle trivial attribuant à toutes les observations du corpus la classe majoritaire (100% ; 76,60% ; 86,75%). Si a fortiori l'on élargit le lexique (e.g. les lexies présentes sur 40% de dialogues), pas un modèle obtenu n'aboutit à une F-mesure inférieure à 90% (toujours pour le succès).

Il semble dès lors inutile de convoquer des métriques autres que les (rappel ; précision ; F-mesure) de la modalité d'échec, notés à présent (r ; p ; F). Il en irait autrement *si la répartition de sortie était davantage balancée*. Exactitude (« accuracy ») et Kappa sont rappelés à titre indicatif, en tant que valeurs symétrisées usuelles de la performance en classification [Japcowicz, 2014].

3 Méthode

L'étude envisage la prédiction de l'échec sous un angle lexical, où la démarche est d'« aplatir » chaque dialogue en un vecteur unique d'une dimension égale à la taille du lexique. Celui-ci est construit selon le mode opératoire qui suit, appelant une précision définitoire.

On appellera TAUX D'OCCURRENCE des entrées lexicales retenues, le pourcentage minimal de dialogues *distincts* où ces entrées doivent être présentes, qu'elles y apparaissent une ou plusieurs fois. Par exemple, l'adjectif « autre » fait partie de la distribution lexicale de Datcha pour un taux d'occurrence de 40%, mais pas de 80%. Il s'agit donc d'un paramètre servant à ajuster le dimensionnement de la distribution lexicale nominale. Les différentes valeurs de taux d'occurrence font l'objet d'un choix d'échelle. Pour opérer un compromis entre combinatoire et observation significative, une échelle de 5% à 75% par incrément de 5% est retenue. La valeur minimale (5%) correspond au lexique maximal, soit en l'occurrence 1141 entrées lexicales non lemmatisées.

Pour chaque entrée, la valeur de la cellule est le nombre d'occurrences dans le dialogue observé. Comme les mots-coordonnées entretiennent des relations de cooccurrence dans le champ linguistique, on choisit de ne pas effectuer de normalisation sur ces valeurs.

Pour un taux d'occurrence fixé, 3 variantes de la distribution nominale sont construites comme décrit ci-après.

- DIFFERENTIATION DES LOCUTEURS : On distingue les entrées lexicales utilisées par T et par C en les préfixant. Par exemple, « bonjour », qui est employé par les deux locuteurs, existera sous les formes « T_bonjour » et « C_bonjour » dans le lexique « différencié ». La distinction des distributions par locuteur cherche à déterminer si une même entrée, observée chez C ou T, traduit la même inclinaison de la conversation vers le succès ou l'échec.

- FILTRAGE DES « MOTS CREUX » : Un filtrage lexical selon une liste de 815 formes fléchies de « mots creux » (« stop words ») est appliqué à la distribution lexicale considérée.

- LEMMATISATION : L'entrée lexicale est le lemme de celle trouvée en corpus. L'outil utilisé est la plateforme « TiLT » (Orange Labs). Par exemple, la flexion « voudrais » trouvée en corpus est représentée par l'entrée lexicale « vouloir ».

On précise que, même en conservant les flexions d'origine, l'outil de segmentation et lemmatisation a une empreinte sur la distribution lexicale, e.g. pour les locutions et les transformations sur les contractions (« au » → « à » + « le »). Afin d'avoir une base homogène de comparaison, *ces transformations sont appliquées aux deux cas*. Ces effets, quoique marginaux, renvoient au débat de fond de la lemmatisation [Brunet, 2000]. Des cas de variantes flexionnelles [Valette & al, 2014] pouvant s'avérer en contexte sémantiquement opposées, étayent une invitation à la vigilance quant aux effets du remplacement par le lexème.

80% des données sont dévolues à l'entraînement avec une validation croisée sur 5 sous-groupes et 20% au test, par tirage au sort. Les modalités étant déséquilibrées mais sans excès (649 observations d'échecs) et les essais de sur-échantillonnage sans effet autre que marginal, les observations ne sont pas corrigées.

L'étude est menée selon le mode opératoire séquentiellement disposé comme suit. On y désigne par « campagne » une phase d'entraînement d'une série de modèles de classification employant différentes valeurs de paramétrage.

- NETTOYAGE DU CORPUS : Pour identifier des phénomènes liés aux entités nommées (EN), le nettoyage de Datcha effectué dans les études antérieures a été poursuivi sur les erreurs résiduelles (nom, téléphone) et sur tous les horodatages. L'exercice a dû définir des limites,

puisque peut être une EN toute « expression linguistique autonome » [Nouvel & al, 2015], pouvant « par ses seules ressources, évoquer un référent ». Ici, les EN non traitées les plus représentatives ont été les adresses postales. Leur diversité toponymique et syntaxique contribue sensiblement à la taille de la distribution lexicale. Mais leur détection et leur remplacement par un terme générique sortait des bornes de l'étude (voir § Perspectives).

- CAMPAGNE GROSSIERE : Cette série de phases d'apprentissage avait pour objet de sélectionner un algorithme de classification au comportement suffisamment et rapidement prometteur : un classifieur bayésien naïf « témoin » et des modèles usuels comme SVM, utilisés dans un contexte analogue mais multimodal par [Salim & al, 2016]. Parmi les algorithmes s'illustrant bien en classification binaire [Torlay, 2017] et [Alpaydin, 2010], le modèle XgBoost a été finalement retenu parmi 5 alternatives.

- CAMPAGNE FINE : Cette série de phases d'apprentissage a exploré de façon systématique le comportement du modèle XgBoost retenu, à chaque palier de taux d'occurrence entre 75% et 5%, pour la distribution lexicale nominale et pour les trois variantes *différenciées* par locuteur ; *filtrée* sur mots creux ; et *lemmatisée*. Soit 15 paliers de taux d'occurrence × (1 + 3) distributions = 60 entraînements-tests.

4 Résultats

Pour réaliser l'ajustement de la fourchette pertinente dans laquelle contraindre la largeur du lexique, on a d'abord observé la variabilité des résultats endogènes à chaque palier du taux d'occurrence.

Ainsi, dans le TABLEAU 1, chaque ligne établit les valeurs moyennes des métriques relatives aux 4 distributions lexicales construites (nominale, différenciée, filtrée, lemmatisée). La colonne 2 indique l'intervalle dans lequel varient leurs tailles respectives. Les 3 colonnes « c.v.(*) » indiquent le *coefficient de variation* (rapport de l'écart-type sur la moyenne) pour le rappel, la précision et la F-mesure, au sein de chaque groupe de 4 distributions, pour un taux d'occurrence donné.

taux occurrence (%)	amplitude Ω(lexique)	rappel moyen	**c.v. (*) rappel**	précision moyenne	**c.v. (*) précision**	F-mesure moyenne	**c.v. (*) F-mesure**
75	4 – 64	69,19	**17,31**	55,58	**18,93**	61,62	**18,15**
70	6 – 74	72,78	**14,59**	64,81	**13,99**	68,53	**14,02**
65	7 – 84	72,42	**18,34**	65,54	**17,83**	68,70	**17,55**
60	12 – 104	75,08	**8,74**	72,58	**8,65**	73,81	**8,65**
55	15 – 118	77,44	**9,54**	71,88	**13,39**	74,53	**11,52**
50	18 – 128	76,73	**7,23**	72,67	**10,44**	74,62	**8,83**
45	21 – 152	77,93	**6,52**	75,19	**8,43**	76,53	**7,50**
40	26 – 182	80,03	**5,33**	77,75	**7,33**	78,86	**6,35**
35	31 – 202	80,97	**2,70**	78,24	**7,14**	79,55	**4,99**
30	37 – 230	83,30	**2,06**	80,28	**7,38**	81,68	**4,09**
25	52 – 274	82,86	**1,80**	80,43	**3,25**	81,60	**1,91**
20	82 – 358	83,71	**1,74**	82,03	**2,29**	82,86	**1,73**
15	117 – 465	82,36	**2,83**	83,48	**0,95**	82,90	**1,04**
10	184 – 645	83,25	**2,17**	84,75	**2,15**	83,99	**1,95**
5	390 – 1141	82,60	**3,50**	83,79	**3,71**	83,18	**3,52**

(*) coefficient de variation

TABLEAU 1 : Variabilités (rappel ; précision ; F-mesure) par palier de taux d'occurrence

L'intervalle [30% ; 50%] apparaît comme zone centrale où la variabilité de la F-mesure est décelable mais comprise entre 4% et 9%, traduisant des résultats bons à médiocres selon le cas (différentiation par locuteur ; filtrage de mots creux ; lemmatisation).

Au-delà, l'amplitude de la variabilité des paliers a tendance à augmenter et le modèle peut rester acceptable comme devenir pire qu'un modèle aléatoire (« zone carencée »). En-deçà [5% ; 25%], on reconnaît un symptôme « d'école » en ce que le modèle n'apprend plus que marginalement de la prise en compte d'un lexique plus large (les mots trop peu fréquents n'améliorent pas le modèle). Mais surtout, les 4 distributions tendent à voir leurs effets respectifs disparaître (« zone saturée »).

Le TABLEAU 2 indique que toutes valeurs paramétriques et tous paliers confondus, le meilleur modèle est trouvé pour un taux d'occurrence de 10% dans le mode de génération du lexique sous forme *lemmatisée*, *indifférenciée par locuteur* et *sans filtrage des mots creux*. Les (r ; p ; F) en sont (85,82% ; 87,05% ; 86,43%). Ancré dans la zone saturée, le modèle ne se situe pas dans un intervalle de confiance permettant de conclure *dans cette zone*, à une influence du paramétrage. L'exactitude (« accuracy ») et le Kappa sont respectivement supérieurs à 93% et 80%, sachant qu'un Kappa supérieur à 70% est usuellement considéré excellent :

Modèle	Ω(lexique)	exactitude	Kappa	rappel	précision	F-mesure
Nominal	324	91,53	77,28	83,21	82,61	82,91
Filtré	184	91,53	77,71	81,94	84,89	83,39
Différencié	645	91,71	77,71	82,01	84,44	83,21
Lemmatisé	295	93,15	81,85	85,82	87,05	86,43

TABLEAU 2 : Détail du palier à 10%, hébergeant le meilleur modèle parmi 60

Hors zone saturée, observons dans le TABLEAU 3 la différence de F-mesure que les modèles différencié, filtré sur mots creux, et lemmatisé, entretiennent respectivement au modèle nominal.

zone	zone de pénurie					zone centrale					zone saturée				
taux occur.	75	70	65	60	55	50	45	40	35	30	25	20	15	10	5
diff vs nom.	6,1	6,0	10,4	3,3	6,5	1,8	3,2	2,2	2,1	0,9	3,7	-0,8	-0,7	0,3	2,0
filt. vs nom.	-17,9	-16,2	-16,7	-10,6	-13,9	-12,9	-9,8	-9,0	-7,0	-6,5	1,5	-3,3	-2,0	0,5	-3,4
lem. vs nom.	5,3	0,8	6,9	2,2	-0,3	-2,6	0,9	-4,7	-0,4	-0,5	2,4	-1,8	-0,5	3,5	3,3

TABLEAU 3 : Différence de F-mesure des 3 variantes de distribution, à la distribution nominale

Le filtrage se traduisant par une réduction de la dimension, il appelle une comparaison spécifique (TABLEAU 4), mesure par mesure, avec le modèle nominal de dimension la plus proche, par équité informationnelle (on prend la plus proche inférieure, qui accentue les conclusions). Les mesures sont comparées en demeurant dans la zone centrale du modèle nominal.

modèle filtré par mots creux					modèle nominal de plus proche dimension inférieure				
taux	Ω(lexique)	rappel	précision	F-mesure	taux	Ω(lexique)	rappel	précision	F-mesure
5	390	78,72	79,86	79,29	10	324	83,21	82,61	82,91
10	184	81,94	84,89	83,39	20	179	84,67	84,06	84,36
15	117	79,45	84,06	81,69	30	115	83,82	82,61	83,21

TABLEAU 4 : Comparaison en F-mesure entre modèle filtré et nominal de dimension inférieure

On discute à présent le comportement des paramètres dans les trois zones instable, centrale et saturée – qui constitue l'éclairage privilégié par la présente étude.

5 Discussion

La différentiation de la distribution améliore les résultats du modèle, en zones d'expression. C'est en zone carencée que son effet est le plus prononcé. La F-mesure y est en moyenne de 6,45% supérieure au modèle nominal (sa meilleure observation y est de 82,96% au taux de 55%). Son effet reste positif en zone centrale mais chute à 2% en moyenne. Il devient indiscernable en zone saturée où l'information purement lexicale se suffit à elle-même. La dynamique à l'œuvre semble indiquer que plus le lexique est large, *moins la connaissance du locuteur apporte à la prédiction*.

L'effet du filtrage des mots creux engage plusieurs lectures. Hors zone saturée, il est en soi naturel que le filtrage de *quelque mot que ce soit* dégrade les performances prédictives (destruction informationnelle). Mais là, le phénomène survient dès la zone centrale (9% de dégradation de F-mesure en moyenne) pour s'accentuer en zone carencée (-15%). La zone saturée présente le même symptôme d'annulation d'effet que précédemment.

En second lieu, il s'agit de comparer la performance du lexique filtré avec un lexique nominal de taille voisine. C'est ce dernier qui l'emporte dans sa propre zone centrale, même en prenant, par convention, la taille inférieure la plus proche. À taille comparable, la part « thématique » du lexique est ainsi *moins indicatrice de l'échec* que les mots réputés creux, ce qui peut constituer le socle légitimant une étude des conditions de généralisation du modèle à d'autres corpus.

La lemmatisation, enfin, a un caractère oscillatoire de moyenne amplitude qui rend a priori caduque toute perspective d'interprétation. Dans la présente structure d'apprentissage de type sacs de mots, elle peut à des paliers adjacents (cf. 75%, 70%, 65%) améliorer sensiblement la performance comme apparaître transparente. En zone centrale, elle peut *dégrader* la performance, suggérant des ambiguïtés de même nature que [Valette & al, 2014] (par exemple, il est plausible qu'un diagnostic de panne occasionnant « des redémarrages » ne soit pas corrélé au succès comme le serait celui occasionnant « un redémarrage » : la confusion flexionnelle serait sur cet exemple génératrice de faux négatifs, si les redémarrages multiples ont une corrélation positive à l'échec).

6 Perspectives

Une combinatoire complète des cas étudiés aurait à court terme vocation à examiner les effets *cumulés* de plusieurs des critères d'apprentissage (e.g. filtrage + différenciation par locuteur). La séquentialité de la dynamique dialogique inspire en fait la mise en œuvre d'approches dédiées. La plus immédiate consisterait à observer la différence de distribution lexicale des locuteurs, de part et d'autre d'un *point d'avancement* du dialogue défini par un *taux arrondi de nombre de tours de parole sur le nombre total de tours de parole du dialogue considéré*. Il constituerait un nouveau paramètre variable du processus d'apprentissage. Une caractérisation fine des comportements séquentiels passe par un changement de la technique d'apprentissage, e.g. au profit de réseaux de neurones récurrents adaptés au texte (LSTM). L'observation unitaire y devient l'entrée lexicale, ou une transformée de type word2vec ou GloVe [Pennington & al, 2014]. Elle est affublée de traits B, I, O délimitant les dialogues et tours de parole. Enfin, dans chaque scénario, une révision de l'état de l'art en reconnaissance d'entités nommées permettrait d'envisager une préparation plus poussée du corpus, réduisant l'inflation lexicale.

Remerciements : Ce travail a été financé partiellement par l'Agence Nationale de la Recherche : ANR-15-CE23-0003 (DATCHA).

Références

ALPAYDIN E. (2010). *Introduction to Machine Learning*. Cambridge, MA: The MIT Press. 431.

BRUNET É. (2000). Qui lemmatise dilemme attise. *Lexicometrica, n°2*.

CLAVEAU V., NCIBI A. (2013). Découverte de connaissances dans les séquences par CRF non-supervisées. *TALN 2013*.

DAMNATI G., GUERRAZ A., CHARLET D. (2016). Web Chat Conversations from Contact Centers: a Descriptive Study, *LREC 2016*.

GRICE H. PAUL (1975). Logic and Conversation. *Syntax and Semantics, Vol. 3. Speech Acts*.

HIRSCHMAN L. (1992). MADCOW: Multi-Site Data Collection for a Spoken Language Corpus.

JAPKOWICZ N., SHAH M. (2014). *Evaluating Learning Algorithms: A Classification Perspective*. Cambridge, MA: The MIT Press. 93.

LOWE R., NOSEWORTHY M., SERBAN I., ANGELARD-GONTIER N., BENGIO Y., PINEAU J. (2017). Towards an Automatic Turing Test: Learning to evaluate dialogue responses. *ICLR 2017*.

NOUVEL D., EHRMANN M., ROSSET S. (2015). *Les entités nommées pour le traitement automatique des langues*. Londres : ISTE Editions.

PENNINGTON J., SOCHER R., MANNING C. D. (2014). GloVe: Global Vectors for Word Representation. *Computer Science Department, Stanford University*.

SALIM S., HERNANDEZ N., MORIN E. (2016). Comparaison d'approches de classification automatique des actes de dialogue dans un corpus de conversations écrites en ligne sur différentes modalités. *TALN 2016*.

TALHA M., BOULAKNADEL S., ABOUTAJDINE D. (2014). RENAM: Système de Reconnaissance des Entités Nommées Amazighes. *TALN 2014*.

TOGNINI-BONELLI, E. (2001). *Corpus linguistics at work*. Amsterdam : John Benjamins.

TORLAY L., PERRONE-BERTOLOTTI M., THOMAS E., BACIU M. (2017). Machine learning – XGBoost analysis of language networks to classify patients with epilepsy. *Brain Informatics Vol. 4, Issue 3*. New York: Springer. 159–169.

VALETTE M., GRABAR N. (2004). Caractérisation de textes à contenu idéologique : statistique textuelle ou extraction de syntagme ? l'exemple du projet PRINCIP. *Le poids des mots, Actes des 7es Journées internationales d'Analyse statistique des Données Textuelles (JADT)*. 1111.

YANG Z., LEVOW G.-A., MENG H. (2012). Predicting User Satisfaction in Spoken Dialog System Evaluation With Collaborative Filtering. *IEEE Journal of Selected Topics in Signal Processing*.

Portée de la négation : détection par apprentissage supervisé en français et portugais brésilien

Clément Dalloux[1] Natalia Grabar[2] Vincent Claveau[1] Claudia Moro[3]
(1) Univ Rennes, Inria, CNRS, IRISA, F-35000 Rennes
(2) UMR 8163 STL CNRS, Université de Lille 3 - France,
(3) Pontifícia Universidade Católica do Paraná (PUC-PR),
(1) prenom.nom@irisa.fr, (2) natalia.grabar@univ-lille3.fr,
(3) c.moro@pucpr.br

RÉSUMÉ
La détection automatique de la négation fait souvent partie des pré-requis dans les systèmes d'extraction d'information, notamment dans le domaine biomédical. Cet article présente nos contributions concernant la détection de la portée de la négation en français et portugais brésilien. Nous présentons d'une part deux corpus principalement constitués d'extraits de protocoles d'essais cliniques en français et portugais brésilien, dédiés aux critères d'inclusion de patients. Les marqueurs de négation et leurs portées y ont été annotés manuellement. Nous présentons d'autre part une approche par réseau de neurones récurrents pour extraire les portées.

ABSTRACT
Negation scope : sequence labeling by supervised learning in French and Brazilian-Portuguese.

Automatic detection of negated content is often a pre-requisite in information extraction systems, especially in the biomedical domain. This paper proposes our contributions on negation scope detection in French and Brazilian Portuguese. We introduce two corpora mainly built with excerpts from clinical trial protocols, describing the inclusion criteria of patients. The corpora have been manually annotated for marking up the negation cues and their scope. Then, we propose a recurrent neural networks approach to acquire the scopes.

MOTS-CLÉS : négation, français, portugais brésilien, apprentissage supervisé, réseau de neurones.

KEYWORDS: negation, French, Brazilian Portuguese, supervised learning, neural network.

1 Introduction

La détection de la négation dans les textes est un des pré-requis qui est utile dans de nombreuses tâches d'extraction d'informations. Dans le domaine biomédical, plusieurs informations existent sous forme non structurée et il y est donc nécessaire de pouvoir différencier, par exemple, l'absence et la présence d'une maladie, ou encore la prise ou non d'un médicament (Chapman *et al.*, 2001; Vincze *et al.*, 2008). Dans le cas d'essais cliniques, ces informations sont déterminantes dans le processus de recrutement des patients car elles indiquent et définissent les critères d'inclusion et d'exclusion de patients pouvant être inclus dans un essai. Il peut s'agit par exemple du statut fumeur ou non d'une personne, de la prise ou non d'un médicament donné, du fait qu'une patiente soit enceinte, etc.

L'article est structuré de la manière suivante. Dans la section qui suit, nous présentons les travaux existants qui proposent des ensembles de données annotées ainsi que des méthodes par systèmes expert ou par apprentissage supervisé pour la détection automatique de la négation et de sa portée. Ces travaux concernent majoritairement le traitement de textes en anglais. Nous présentons ensuite, dans la section 3, les ensembles de données que nous avons annotées, en français et portugais brésilien, dans le but d'automatiser la détection de la négation dans d'autres langues que l'anglais. Notre méthode de détection de la portée par apprentissage supervisé est présentée dans la section 4. Dans la section 5, nous présentons les résultats de notre méthode sur ces ensembles de données, ainsi qu'une analyse des erreurs les plus fréquentes. Enfin, nous présentons nos conclusions et perspectives dans la dernière section.

2 Travaux existants

Dans (Dalloux, 2017), nous avons présenté une grande partie des travaux pertinents pour cette tâche. Dans cette section, nous les décrivons brièvement et complétons notre état de l'art.

2.1 Les données

Ces dernières années, souvent dans le cadre de *shared tasks*, les énoncés/assertions négatifs ont été annotés dans plusieurs corpus de spécialité. On peut y distinguer deux catégories : (1) les corpus avec les marqueurs et leurs portées annotés et (2) les corpus se focalisant sur les concepts et entités nommées.

Dans la première catégorie, on retrouve Bioscope (Vincze *et al.*, 2008), qui est composé de comptes-rendus d'examens radiologiques, d'articles scientifiques ainsi que de résumés d'articles annotés avec l'incertitude et la négation. En tout, le corpus comprend 20 924 phrases dont environ 13 % sont négatives. Les ensembles de données de *SEM-2012 [1] sont annotés de la même manière. Ce corpus est composé d'un roman et de trois nouvelles de Sherlock Holmes par Conan Doyle et comprend 5 520 phrases, dont 1 227 négatives. Les marqueurs de la négation sont constitués soit d'un préfixe soit d'un ou de plusieurs mots qui modifient, sauf exceptions, la polarité et donc le sens de la phrase. La portée est l'effet du marqueur, qui s'étend sur toute la phrase ou sur une partie de cette phrase. Dans cette catégorie de travaux, les marqueurs de la négation sont donc ciblés.

Les travaux de la seconde catégorie se focalisent sur le contexte entourant les entités nommées. Par exemple, la compétition i2b2/VA-2010 (Uzuner *et al.*, 2011) présentait trois tâches, dont la classification d'assertions se focalisant sur l'attribution des types d'assertions pour les concepts médicaux. En d'autres termes, à chaque concept était attribué l'une de ces 6 classes d'assertions : *present*, *absent*, *possible*, *conditional*, *hypothetical* ou *not associated with the patient*. Dans cette catégorie de travaux, ce sont les entités nommées qui sont les pivots des études.

Mipacq (Albright *et al.*, 2013) est un autre exemple de corpus, qui est constitué de données cliniques annotées avec plusieurs couches d'étiquettes syntaxiques et sémantiques. Dans ce corpus, chaque entité UMLS dispose de deux emplacements d'attributs : *Negation*, qui peut prendre deux valeurs : *true* ou *false*, et *status*, qui peut être *none*, *possible*, *HistoryOf* ou *FamilyHistoryOf*.

1. https://www.clips.uantwerpen.be/sem2012-st-neg/

2.2 Détection automatique

D'une manière générale, il existe deux familles d'approches pour aborder la détection automatique de la négation. La première famille couvre la conception de systèmes experts, tels que *NegEx* (Chapman *et al.*, 2001) et son adaptation française (Deléger & Grouin, 2012), *Negfinder* (Mutalik *et al.*, 2001), ou bien *ConText* (Harkema *et al.*, 2009) et son adaptation française (Abdaoui *et al.*, 2017). Notons que le système *ConText* a une portée plus large et permet de détecter la négation, la temporalité et le sujet dans les textes cliniques. Très récemment, Peng *et al.* (2018) ont proposé *NegBio* dont le code est disponible en ligne [2]. Leur système repose sur des règles définies à partir des graphes de dépendances (*universal dependency graph* (UDG)).

La deuxième famille rassemble de nombreux travaux qui utilisent la classification par apprentissage supervisé à l'aide de méthodes telles que les champs aléatoires conditionnels (*Conditional Random Fields* ou CRFs), les machines à vecteurs support (SVM) ou les réseaux de neurones (Velldal *et al.*, 2012; Read *et al.*, 2012; Packard *et al.*, 2014; Fancellu *et al.*, 2016). Enfin, dans (Dalloux *et al.*, 2017), nous proposions déjà plusieurs approches par réseau de neurones et comparions leurs résultats pour le français et l'anglais.

Dans cet article, nous publions les résultats obtenus avec la version la plus récente de notre corpus, ainsi qu'avec un corpus en portugais brésilien, une langue rarement représentée dans les travaux de TAL.

3 Nos données

Nous présentons dans cette section deux corpus, l'un en français (FR), l'autre en portugais brésilien (PTBR), annotés manuellement pour marquer les informations liées à la négation : les marqueurs et leur portée. Ces corpus sont similaires car élaborés à partir de protocoles d'essais cliniques dans les deux langues. Le corpus français a été annoté par l'équipe de l'IRISA d'INRIA, tandis que le corpus brésilien a été annoté par trois étudiants de l'école de médecine de l'Université pontificale catholique du Paraná [3].

Les protocoles en français sont issus du registre des essais cliniques de l'hôpital Gustave Roussy, ainsi que de l'Institut National du Cancer, auxquels nous avons ajouté quelques cas cliniques. Le corpus français est pré-traité avec TreeTagger (Schmid, 1994) pour effectuer l'étiquetage morpho-syntaxique et la lemmatisation. Les protocoles en portugais brésilien sont issus du site brésilien dédié aux essais cliniques [4]. L'étiquetage morpho-syntaxique (*Universal POS tags*) et les lemmes du corpus portugais brésilien sont obtenus à l'aide de *RDRPOSTagger* proposé par Nguyen *et al.* (2015).

Dans les exemples qui suivent, les marqueurs de négation sont en gras et leur portée est marquée entre crochets. Le rationale des annotations diffère légèrement dans les deux langues. Si l'annotation des marqueurs est identique, il existe des différences dans l'annotation de leur portée : dans le corpus brésilien (exemples 4, 5, 6) la portée se limite souvent aux groupes nominaux tandis qu'en français (exemples 1, 2, 3) toutes les séquences (groupes nominaux, propositions, subordonnées...) impactées par le marqueur font partie de la portée. Ainsi, dans l'exemple 4, *apresentar* n'est pas annoté, ni

2. https://github.com/ncbi-nlp/NegBio
3. https ://www.pucpr.br/
4. http://ensaiosclinicos.gov.br/

usuários de dans l'exemple 6.

1. **absence** [de ganglion métastatique]
2. En cas d'**in**[opérabilité] et/ou **im**[possibilité de réirradier],...
3. ...[des patients éligibles atteints d'une tumeur maligne et porteurs de la mutation BRAFV600] ayant précédemment été inclus et traités dans un protocole antérieur portant sur le Vemurafénib et **n'**[ayant] **pas** [satisfait aux critères du protocole sur la progression de la maladie]...
4. Grupo Controle : **Não** apresentar [DTM].
5. **Ausência de** [evidência clínica de imunossupressão].
6. **não** usuários de [drogas que causem dependência química].

Comme nous pouvons voir, la négation peut être exprimée de différentes manières : morphologique (*im-*), lexicale (*absence, ausência*) et grammaticale (*ne pas, não*).

Dans le tableau 1, nous présentons quelques statistiques sur nos corpus : le nombre de mots, la variété du vocabulaire, le nombre de phrases et le nombre de phrases négatives.

	FR	PTBR
Mots	134 386	48 204
Vocabulaire	8 133	6 453
Phrases	5 394	3 228
Phrases négatives	820 (15,20 %)	640 (19,83 %)

TABLEAU 1 – Statistique des corpus constitués pour notre travail

Dans le tableau 2, nous présentons des exemples de phrases en français et en brésilien annotées avec la négation et sa portée. Les champs renseignés sont : l'identifiant unique par phrase (*#Phrase*), la position de chaque token dans la phrase (*Position*), les différents descripteurs linguistiques (*Forme*, *Lemme*, *POS-tag*), ainsi le marqueur de la négation et sa portée. Le dernier exemple du tableau est issu du corpus brésilien. Comme d'autres phrases de ce corpus, il ne contient qu'un seul mot : *gestantes* signifiant que l'étude porte sur les femmes enceintes. Les phrases affirmatives sont indiqués par *** dans la sixième colonne.

4 Méthodes

Nous abordons la tâche de détection de la portée de la négation comme une tâche de classification binaire (*Inside-Outside*). Les informations recherchées doivent être prédites grâce à l'algorithme d'apprentissage supervisé, que nous décrivons dans la suite de cette section.

Inspirée de Fancellu *et al.* (2016), notre approche, dédiée à la détection de la portée de la négation, utilise un réseau de neurones Long Short-Term Memory (LSTM) bidirectionnel suivi, en sortie, d'une prédiction par champs aléatoires conditionnels (CRF). Un LSTM bidirectionnel est la combinaison de deux structures de réseaux de neurones récurrents (RNN) : le réseau de neurones récurrent bidirectionnel proposé par Schuster & Paliwal (1997), qui opère dans le sens de lecture et dans le sens contraire. La passe arrière est particulièrement importante dans le cas de la détection de la portée puisque les unités affectées peuvent également se trouver avant le marqueur. Le LSTM proposé par

#Phrase	Position	Forme	Lemme	POS-tag	Marqueur	Portée
5105	0	L’	le	DET :ART	_	_
5105	1	abdomen	abdomen	NOM	_	abdomen
5105	2	est	être	VER :pres	_	_
5105	3	souple	souple	ADJ	_	_
5105	4	et	et	KON	_	_
5105	5	sans	sans	PRP	sans	_
5105	6	défense	défense	NOM	_	défense
5105	7	.	.	SENT	_	_
2247	0	déficit	déficit	NOUN	_	_
2247	1	visual	visual	ADJ	_	_
2247	2	grave	grav	ADJ	_	_
2247	3	sem	sem	ADP	sem	_
2247	4	correção	correçã	NOUN	_	correção
2247	5	.	.	PUNCT	_	_
2916	0	Gestantes	gestant	ADJ	***	
2916	1	.	.	PUNCT	***	

TABLEAU 2 – Extraits des corpus annotés en français et brésilien

Hochreiter & Schmidhuber (1997) est également plus efficace que le RNN classique pour apprendre des dépendances de long-terme. Les CRFs (Lafferty *et al.*, 2001), modèles statistiques souvent utilisés pour des données séquentielles, semblent être particulièrement efficaces pour l’étiquetage de séquences dans les textes. Nous les préférons donc à la couche de prédiction la plus courante, la couche *softmax*.

Implémenté à l’aide de *Tensorflow* (Abadi *et al.*, 2016), notre système prend en entrée une instance $I(n, c, t)$, où chaque mot est représenté par un vecteur n (*word-embedding*), un vecteur c, qui détermine si le mot fait partie d’un marqueur (*cue-embedding*), ainsi que d’un vecteur t, qui est la représentation vectorielle de l’étiquetage morpho-syntaxique pour chaque mot (*postag-embedding*).

Pour chaque système, nous utilisons des paramètres d’entraînement définis de façon empirique. Nos embeddings sont de dimension $k = 50$. La couche cachée compte 200 unités (400 pour le BiLSTM, qui nécessite deux couches cachées concaténées). 50 périodes d’entraînement permettent d’atteindre le meilleur score F_1 sur l’ensemble de validation.

Les corpus sont segmentés en trois parties : 60% pour l’ensemble d’entraînement, 15% pour l’ensemble de validation et 25% pour l’ensemble de test. Les résultats obtenus sur le corpus de test sont évalués contre les données de référence avec les mesures d’évaluation classiques : la précision P, qui quantifie la pertinence de l’étiquetage, le rappel R, qui quantifie la sensibilité de l’étiquetage, ainsi que la moyenne harmonique de la précision et du rappel noté F_1.

5 Résultats et discussion

Dans le tableau 3, nous présentons les résultats de notre approche sur nos deux corpus. Sont indiqués les résultats sur les ensembles de validation et de test. À titre de comparaison, nous indiquons

Dataset	Mots étiquetés			Portées exactes		
	P	R	F_1	P	R	F_1
français-valid	93,72	86,22	89,81	100	72,48	84,04
français-test	88,29	84,68	86,45	100	53,55	69,75
PTBR-valid	79,06	69,93	74,22	100	33,71	50,42
PTBR-test	77,61	64,58	70,50	100	34,29	51,06
SEM-2012	91,24	87,10	**89,12**	100	62,5	76,92
*SEM-2012**	92,62	85,13	88,72	99,40	63,87	**77,7**

TABLEAU 3 – Résultats de notre système sur nos corpus. Les résultats sont donnés en pourcentage. (Dalloux *et al.*, 2017)*, (Fancellu *et al.*, 2016)**

également les résultats que nous obtenons sur le corpus de *SEM-2012, ainsi que ceux de Fancellu *et al.* (2016) obtenus sur ce même corpus. Bien que l'ensemble d'entraînement français soit plus petit (d'environ 40 %), nous obtenons des résultats proches de ceux obtenus sur le corpus en anglais, avec un score F_1 de 86,45 en français contre 89,12 en anglais. Les résultats sont bien moins convainquants sur le corpus brésilien, qui est encore plus petit (d'environ 55 % pour l'ensemble d'entraînement). Il nous semble que les différences dans la façon d'annoter ont également une influence importante sur les résultats, l'annotation en français se rapprochant bien plus de celle de *SEM-2012 qu'en brésilien. Notons que la précision est privilégiée par le système, avec des valeurs proches ou égales à 100.

Un examen des résultats permet d'isoler des cas récurrents d'erreurs. Pour le corpus français, nous pouvons ainsi noter l'omission de groupes verbaux et l'inclusion d'incises :

— GOLD : Dans cet essai, **ni** [le patient,] **ni** [le médecin] **ne** [connaîtront quel traitement] (ProCervix ou placebo) [est administré].

— PRED : Dans cet essai, **ni** [le patient], **ni** [le médecin] **ne** [connaîtront quel traitement] (ProCervix ou placebo) est administré.

— GOLD : Par ailleurs [les patientes du second groupe ayant un risque de rechute potentiellement bas] (grade génomique bas) **ne** [recevront] **pas** [de chimiothérapie].

— PRED : Par ailleurs les patientes du second groupe ayant [un risque de rechute potentiellement bas] ([grade génomique bas)] **ne** [recevront] **pas** [de chimiothérapie].

Dans le corpus brésilien, les cas d'erreurs s'avèrent plus complexes. En effet, nous constatons qu'un grand nombre de prédictions combinent des erreurs de précision et de rappel au sein d'une même phrase. Les exemples ci-dessous illustrent cette situation :

— GOLD : **Ausencia** de diagnóstico de [doenças neuromusculares], [trauma], [tumores] ou [abscessos raquimedulares], [hemiplegia]/ [paresia], [lesão de plexo] ou [encefalopatia cerebral].

— PRED : **Ausencia** de diagnóstico [de doenças neuromusculares], [trauma], tumores ou [abscessos raquimedulares], hemiplegia/paresia, lesão de plexo ou [encefalopatia] cerebral.

— GOLD : que **não** apresentem outras [doenças neurológicas] ou [ortopédicas diagnosticadas].

— PRED : que **não** apresentem [outras doenças neurológicas] ou [ortopédicas] diagnosticadas.

Une des erreurs concerne l'inclusion ou non de déterminants (*de, outras*) dans la portée des marqueurs. Une autre erreur concerne la complétude de groupes nominaux (*encefalopatia cerebral* et *ortopédicas diagnosticadas*). Il s'agit sans doute des annotations non homogènes, qui peuvent être corrigées assez facilement.

6 Conclusions et perspectives

Dans cet article, nous proposons deux types de contributions.

Une première contribution concerne deux nouveaux corpus de données biomédicales, en français et en brésilien, annotées avec les informations sur la négation (les marqueurs et leur portée). Les corpus seront finalisés grâce à l'intégration de données nouvelles provenant d'autres sources, telles que des articles scientifiques ou bien des cas cliniques. L'accord inter-annotateur est en cours de calcul, s'en suivra une étape d'harmonisation des annotations.

La seconde contribution de notre travail consiste en l'exploitation d'un LSTM bidirectionnel pour la détection automatique de la portée de la négation. Les expériences sont effectuées et évaluées dans deux langues qui n'ont pas connu beaucoup de travaux de ce type : en portugais brésilien et en français. Sur le corpus français, notre système montre des performances qui s'approchent de celles de Packard *et al.* (2014) et Fancellu *et al.* (2016) sur les données de **SEM-2012*, bien que le contenu de nos corpus soit très différent et qu'il nous semble y trouver plus de variations dans la structure de phrases négatives (énumérations, grandes portées discontinues, etc.).

Néanmoins, il nous reste plusieurs points à explorer. En effet, nous souhaitons exploiter différentes architectures de réseaux de neurones récurrents, ainsi que différents types de *word embeddings*. Nous prévoyons également de comparer nos résultats sur le français avec des systèmes experts tels que ceux proposés par Deléger & Grouin (2012) ou bien par (Abdaoui *et al.*, 2017).

Une première version de notre système est désormais accessible et utilisable en ligne à cette adresse : `https://allgo.inria.fr/webapps/173`

Remerciements

Ce travail a bénéficié d'une aide de l'état attribuée au labex COMIN LABS et gérée par l'Agence Nationale de la Recherche au titre du programme « Investissements d'avenir » portant la référence ANR-10-LABX-07-01.

Nous remercions également les relecteurs pour les remarques constructives qui ont permis d'améliorer la présentation de notre travail.

Références

ABADI M., AGARWAL A., BARHAM P., BREVDO E., CHEN Z., CITRO C., CORRADO G. S., DAVIS A., DEAN J., DEVIN M. & ET AL. (2016). Tensorflow : Large-scale machine learning on heterogeneous distributed systems. *arXiv preprint arXiv :1603.04467*.

ABDAOUI A., TCHECHMEDJIEV A., DIGAN W., BRINGAY S. & JONQUET C. (2017). *French ConText : Détecter la négation, la temporalité et le sujet dans les textes cliniques Français*, In *4e édition du Symposium sur l'Ingénierie de l'Information Médicale*.

ALBRIGHT D., LANFRANCHI A., FREDRIKSEN A., STYLER IV W. F., WARNER C., HWANG J. D., CHOI J. D., DLIGACH D., NIELSEN R. D., MARTIN J. *et al.* (2013). Towards comprehensive

syntactic and semantic annotations of the clinical narrative. *Journal of the American Medical Informatics Association*, **20**(5), 922–930.

CHAPMAN W. W., BRIDEWELL W., HANBURY P., COOPER G. F. & BUCHANAN B. G. (2001). A simple algorithm for identifying negated findings and diseases in discharge summaries. *Journal of Biomedical Informatics*, **34**(5).

DALLOUX C. (2017). Détection de l'incertitude et de la négation : un état de l'art. In *19es REncontres jeunes Chercheurs en Informatique pour le TAL (RECITAL 2017)*, p. 94–107.

DALLOUX C., CLAVEAU V. & GRABAR N. (2017). Détection de la négation : corpus français et apprentissage supervisé. In *SIIM 2017 - Symposium sur l'Ingénierie de l'Information Médicale*, p. 1–8, Toulouse, France.

DELÉGER L. & GROUIN C. (2012). Detecting negation of medical problems in french clinical notes. In *Proceedings of the 2nd ACM SIGHIT International Health Informatics Symposium*.

FANCELLU F., LOPEZ A. & WEBBER B. (2016). Neural networks for negation scope detection. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*, volume 1.

HARKEMA H., DOWLING J. N., THORNBLADE T. & CHAPMAN W. W. (2009). Context : an algorithm for determining negation, experiencer, and temporal status from clinical reports. *Journal of biomedical informatics*, **42**(5), 839–851.

HOCHREITER S. & SCHMIDHUBER J. (1997). Long short-term memory. *Neural Comput.*, **9**(8).

LAFFERTY J., MCCALLUM A., PEREIRA F. *et al.* (2001). Conditional random fields : Probabilistic models for segmenting and labeling sequence data. In *Proceedings of the eighteenth international conference on machine learning, ICML*, volume 1.

MUTALIK P. G., DESHPANDE A. & NADKARNI P. M. (2001). Use of general-purpose negation detection to augment concept indexing of medical documents : a quantitative study using the umls. *Journal of the American Medical Informatics Association : JAMIA*, **8**(6).

NGUYEN D. Q., NGUYEN D. Q., PHAM D. D. & PHAM S. B. (2015). A robust transformation-based learning approach using ripple down rules for part-of-speech tagging. *AI Communications*, **29**(3), 409–422.

PACKARD W., BENDER E. M., READ J., OEPEN S. & DRIDAN R. (2014). *Simple Negation Scope Resolution through Deep Parsing : A Semantic Solution to a Semantic Problem.*, In *ACL (1)*.

PENG Y., WANG X., LU L., BAGHERI M., SUMMERS R. & LU Z. (2018). Negbio : a high-performance tool for negation and uncertainty detection in radiology reports. *AMIA 2018 Informatics Summit*.

READ J., VELLDAL E., ØVRELID L. & OEPEN S. (2012). *Uio 1 : Constituent-based discriminative ranking for negation resolution*, In *Proceedings of the First Joint Conference on Lexical and Computational Semantics-Volume 1 : Proceedings of the main conference and the shared task, and Volume 2 : Proceedings of the Sixth International Workshop on Semantic Evaluation*. Association for Computational Linguistics.

SCHMID H. (1994). Probabilistic part-ofispeech tagging using decision trees. In *Proceedings of International Conference on New Methods in Language Processing, Manchester, UK*.

SCHUSTER M. & PALIWAL K. K. (1997). Bidirectional recurrent neural networks. *IEEE Transactions on Signal Processing*, **45**(11).

UZUNER Ö., SOUTH B. R., SHEN S. & DUVALL S. L. (2011). 2010 i2b2/va challenge on concepts, assertions, and relations in clinical text. *Journal of the American Medical Informatics Association*, **18**(5), 552–556.

VELLDAL E., ØVRELID L., READ J. & OEPEN S. (2012). Speculation and negation : Rules, rankers, and the role of syntax. *Computational Linguistics*, **38**(2).

VINCZE V., SZARVAS G., FARKAS R., MÓRA G. & CSIRIK J. (2008). The bioscope corpus : biomedical texts annotated for uncertainty, negation and their scopes. *BMC Bioinformatics*, **9**.

Le corpus PASTEL pour le traitement automatique de cours magistraux

Salima Mdhaffar Antoine Laurent Yannick Estève
Laboratoire d'Informatique de l'Université du Mans (LIUM), Avenue Laennec, Le Mans, France
prenom.nom@univ-lemans.fr

RÉSUMÉ

Le projet PASTEL étudie l'acceptabilité et l'utilisabilité des transcriptions automatiques dans le cadre d'enseignements magistraux. Il s'agit d'outiller les apprenants pour enrichir de manière synchrone et automatique les informations auxquelles ils peuvent avoir accès durant la séance. Cet enrichissement s'appuie sur des traitements automatiques du langage naturel effectués sur les transcriptions automatiques. Nous présentons dans cet article un travail portant sur l'annotation d'enregistrements de cours magistraux enregistrés dans le cadre du projet CominOpenCourseware. Ces annotations visent à effectuer des expériences de transcription automatique, segmentation thématique, appariement automatique en temps réel avec des ressources externes... Ce corpus comprend plus de neuf heures de parole annotées. Nous présentons également des expériences préliminaires réalisées pour évaluer l'adaptation automatique de notre système de reconnaissance de la parole.

ABSTRACT

PASTEL corpus for automatic processing of lectures

PASTEL project studies the acceptability and the usability of automatic transcriptions for lectures. The aim of this project is to provide to learners a synchronously and automatically enrichment of the information that they can access during the session. This enrichment is based on automatic processing of natural language performed on automatic transcriptions. In this article, we present the annotation of lectures recorded in the context of the CominOpenCourseware project. These annotations aim to perform experiments of automatic transcription, thematic segmentation, automatic real-time matching with external resources...This corpus includes more than nine hours of annotated speech. Preliminary experiments are conducted to evaluate speech quality in our speech recognition system.

MOTS-CLÉS : Corpus, Transcription, Annotation, Segmentation Automatique, Enrichissement Automatique, Système de Reconnaissance de la Parole, Adaptation du Modèle de Langage.

KEYWORDS: Corpus, Transcription, Annotation, Automatic Segmentation, Automatic Enrichment, Speech Recognition System, Language Model Adaptation.

1 Introduction

Depuis la démocratisation des technologies de l'information et de la communication et leur interpénétration de plus en plus large et profonde avec les activités humaines, le monde de l'enseignement supérieur et de la formation pour adultes est de plus en plus interrogé par la société quant au renouvellement et à l'adaptation des pratiques pédagogiques. D'une part, les frontières entre apprentissage guidé et auto-apprentissage sont de moins en moins marquées, ce qui tend à la redéfinition du rôle de l'enseignant et de l'apprenant, et d'autre part la technologie, de plus en plus accessible, permet de diversifier et de combiner les modes d'interaction enseignant/apprenant et apprenant/apprenant.

Les technologies de reconnaissance de la parole approchent d'un niveau de maturité suffisant qui permet d'envisager de nouvelles possibilités au niveau de l'instrumentation des pratiques pédagogiques et générer de nouveaux usages. Par définition, la reconnaissance automatique de la parole vise à transcrire un énoncé produit oralement. Ces transcriptions automatiques peuvent alors être l'objet de traitements automatiques du langage naturel à l'aide de procédés généralement éprouvés sur du langage écrit.

Sur la base de transcriptions automatiques, il est possible de faciliter les échanges entre les apprenants ou entre tuteurs et apprenants, en s'appuyant sur ces transcriptions pour le développement d'outils pour la prise de notes individuelle et collaborative, ou la rédaction de compte-rendu, et pour la délimitation de zones particulièrement intéressantes du discours qui peuvent être gérées par le tuteur en direct ou a posteriori : zones traitant de notions mal comprises, demandes d'approfondissement ou d'exemple, etc.

Ainsi, le projet PASTEL a pour objectif d'explorer le potentiel de la transcription automatique en temps réel pour l'instrumentation de situations pédagogiques mixtes, où les modalités d'interaction sont présentielles ou à distance, synchrones ou asynchrones.

Différentes recherches dans la littérature ont démontré les avantages de développer des applications de traitement automatique des langues (TALN) dans le cadre pédagogique à savoir les travaux de (Ho *et al.*, 2005), (Hwang *et al.*, 2012) et (Shadiev *et al.*, 2014).

Dans cet article, nous décrivons le corpus que nous avons réalisé dans le cadre de ce projet, ainsi que des expériences préliminaires. Ce corpus composé de transcriptions manuelles du discours d'enseignement en situation de cours magistraux. Il s'accompagne d'une segmentation manuelle. Les données et les annotations seront distribuées sous licence libre à la communauté.

Ce corpus va aider au développement et à l'expérimentation d'une application dans un cadre pédagogique, va permettre l'évaluation des systèmes développés et des approches proposées, et va apporter à la communauté de recherche en TALN un corpus dédié au domaine éducatif.

Cet article détaille la création et l'utilisation du corpus. Il est structuré comme suit : la deuxième section présente les motivations de la création du corpus. La section 3 présente le corpus. La section 4 décrit un exemple d'utilisation de ce corpus. Enfin, la section 5 présente la conclusion.

2 Motivation de la création du corpus

La disponibilité d'un corpus de discours d'enseignement transcrit et segmenté manuellement offre plusieurs possibilités applicatives. Nous citons dans cette section les motivations qui nous ont conduit

à construire le corpus présenté dans cet article.

2.1 Adaptation automatique de modèles de langage

Le modèle de langage (ML) est l'un des constituants d'un système de reconnaissance de la parole. Un ML a pour but de guider le décodeur à choisir la séquence de mots la plus probable. Il est généralement formé à partir d'une large quantité de données représentatives de la tâche pour laquelle le modèle sera utilisé. Cependant, ce ML n'est pas fiable lorsqu'il s'agit de transcrire des documents oraux traitants d'une autre tâche. Les systèmes dont les modèles de langage sont entraînés à partir de données généralistes ne sont pas performants pour transcrire des données liées à des domaines spécifiques. Les données d'apprentissage d'un modèle de langage sont sous forme de séquences de mots collectés à partir d'un discours réel qui sont transcrits manuellement ou semi-automatiquement. Par conséquent, la construction d'un nouveau ML pour chaque domaine est très coûteuse. L'idée classique est d'utiliser un modèle de langage généraliste et de l'adapter aux données du domaine. Dans le cadre de ce travail, où il s'agit de transcrire des cours magistraux portant sur divers domaines, il est donc nécessaire d'avoir recours à des techniques d'auto-adaptation de ces modèles. L'une de ces techniques, présentée dans cet article, sera évaluée grâce au corpus que nous présentons.

2.2 Segmentation et enrichissement de contenu pédagogique

Les transcriptions automatiques peuvent également servir pour la recherche automatique de matériaux pédagogiques et d'informations complémentaires. En travaillant à la structuration automatique du discours du tuteur, par exemple en découpant ce discours en segments thématiques, il est possible d'extraire de ces segments des mots clés, ou de caractériser ces segments par d'autres moyens, de manière telle qu'il soit possible de lier un segment thématique à un ensemble de sources d'informations disponibles par ailleurs et non produites par le tuteur. Dans un contexte de cours magistral, en exploitant si possible les sources d'informations a minima fournies par l'enseignant, il s'agira de segmenter à la volée, c'est-à-dire en direct au fur et à mesure de la transcription automatique en temps réel, les sorties d'un système de reconnaissance automatique de la parole appliqué sur le discours de l'enseignant. Cette segmentation sera de type thématique : il s'agira de détecter les frontières de zones homogènes au niveau du contenu, qu'il faudra caractériser afin de le lier avec des documents pédagogiques disponibles dans une base de connaissances extérieure (documents extérieurs, issus de bases de données spécifiques, d'encyclopédies en ligne comme Wikipédia, ou d'autres sources du web). Dans la littérature, la tâche de la segmentation thématique est relativement bien explorée en TALN, à travers par exemple les travaux de (Bouchekif *et al.*, 2015), (Galley *et al.*, 2003), (Caillet *et al.*, 2004). Dans ces travaux, la segmentation thématique a généralement été réalisée sur des documents finalisés, et non en temps réel. La réalisation de cette tâche, en particulier pour une approche par apprentissage automatique, nécessite un corpus segmenté thématiquement.

2.3 Obtention d'un découpage de séquences autonomes

Cette tâche vise à découper le cours en une suite de séquences de quelques minutes homogènes, de manière chaque séquence puisse être considérée comme un chapitre du cours. Il s'agit d'être capable de reconstruire le plan du cours. Un niveau de granularité plus fin est également pris en compte, qui consiste à segmenter le cours en zone de description de concepts, dont les frontières peuvent être

délimitées par la synchronisation avec une ou plusieurs diapositives dans le cas de cours magistraux qui en utilisent. Le corpus PASTEL est ainsi annoté en plusieurs niveaux de granularité.

3 Corpus

Cette section présente les données, les conventions d'annotation ainsi que les statistiques du corpus.

3.1 Origine du corpus

Les données utilisées sont issus du projet CominOpenCourseware (COCo) [1] qui met à disposition un certain nombre de vidéos avec des ressources potentielles. Les vidéos collectées concernent des cours magistraux de niveau licence. Les vidéos sont alignées temporellement avec les diapositives de la présentation.

Ainsi quelques vidéos sont issues de la plateforme Umotion [2] : plateforme vidéo de l'Université du Mans.

3.2 Annotations

Il est très important pour la qualité et la fiabilité des annotations que des conventions soit mises en place pour guider les annotateurs humains.

Transcription

Pour la partie de la transcription, le logiciel Transcriber [3] a été utilisé. Transcriber est un outil d'aide à l'annotation manuelle de signal vocal (Barras *et al.*, 2001). Il fournit une interface utilisateur graphique conviviale pour segmenter les enregistrements vocaux de longue durée, les transcrire, et marquer les tours de parole, les changements de sujets et les conditions acoustiques. Les conventions généralement utilisées pour les transcriptions de campagnes d'évaluation (Gravier *et al.*, 2004) ont servi de guide pour transcrire les cours enregistrés. Pour accélérer le traitement, la transcription de référence du corpus PASTEL a été réalisée de manière semi-automatique. Une première transcription est générée par un système de reconnaissance de la parole générique (dont le modèle de langage ne contient pas des données du domaine) avant d'être corrigé par l'annotateur.

Segmentation

Afin de guider la segmentation thématique, nous devons répondre à la question suivante : " Qu'est-ce qu'un sous-thème ? " dans un cours qui est monothématique (l'objet principal du cours). Nous avons décidé de répondre à cette question en fonction des motivations que nous avons énumérées dans la section précédente. Nous sommes partis de l'hypothèse qu'une frontière thématique ne peut se situer qu'au voisinage d'un changement de diapositive pendant le cours. Par conséquent, à chaque changement de diapositive, il est nécessaire d'annoter :

1. s'il y a un changement thématique ou non,
2. l'instant exact du changement thématique défini comme étant positionné entre deux mots,

1. http ://www.comin-ocw.org/
2. umotion.univ-lemans.fr/
3. http ://trans.sourceforge.net/en/presentation.php

3. la granularité du changement thématique (1 ou 2).

La granularité 1 est utilisée pour marquer qu'une nouvelle notion est abordée tout en restant dans le même sous-thème. La granularité 2 est utilisée lorsqu'il y a un changement de sous-thème plus général qui permet d'arrêter l'apprentissage à ce moment-là et de reprendre plus tard l'apprentissage d'autres notions. La granularité 2 permet ainsi de chapitrer le cours, chaque chapitre étant constitué de segments de type 1.

Nous avons aussi ajouté à l'annotation la notion d'interruption : il s'agit de localiser les portions de discours qui correspondent à des moments de gestion de l'attention du public, de problèmes techniques (gérer un problème de vidéo-projecteur par exemple), etc.

L'annotation a été effectuée par deux annotateurs étudiants en master en linguistique à l'aide de l'outil ELAN [4] (Auer *et al.*, 2010). ELAN (**E**UDICO **L**inguistic **A**nnotator) est un outil d'annotation qui permet de créer, éditer, visualiser et rechercher des annotations pour des données vidéo et audio. Chaque annotateur a annoté le corpus conformément aux directives précisées dans la section précédente. Par la suite, les deux annotateurs ont confronté leurs annotations pour trouver un consensus et proposer une annotation finale.

3.3 Statistiques du corpus

Le corpus annoté comprend neuf cours. La durée totale du corpus est d'environ 9 heures. Le tableau 1 illustre quelques statistiques de notre corpus. La deuxième, la troisième et la quatrième colonnes du tableau représentent respectivement le nombre de segments de granularité 1, le nombre de segments de granularité 2 et le nombre de segments de type "interruption". La dernière colonne contient la durée de chaque cours.

Nom du cours	**Gran. 1**	**Gran. 2**	**Interp**	**Durée**
Introduction à l'informatique	31	2	2	1h 04mn 42s
Introduction à l'algorithmique	38	10	3	1h 17mn 28s
Les fonctions dans l'algorithmique	35	3	3	1h 14mn 29s
Réseau sociaux et graphes	43	7	7	1h 05mn 51s
Algoritmique distribuée	72	5	3	1h 16mn 30s
Langage naturel	52	5	5	1h 09mn 35s
Architecture république	49	7	0	1h 21mn 14s
Méthode traditionnelle	12	7	1	0h 41mn 02s
Imagerie	57	0	1	1h 08mn 14s

TABLE 1 – Statistiques du corpus (Interp : Interruption, Gran : Granularité)

4 Adaptation du modèle de langage

Cette section présente les premiers travaux menés dans le projet PASTEL pour l'adaptation du modèle de langage.

4. https ://tla.mpi.nl/tools/tla-tools/elan/

4.1 Expérimentation

Les expériences effectuées pour l'adaptation du modèle de langage s'inspirent du travail présenté dans (Lecorvé *et al.*, 2008). Le processus d'adaptation que nous avons effectué dans ce travail consiste à :

1. Identifier les mots-clés en utilisant le critère TF-IDF à partir des diapositives du cours. Cette étape est précédée par un prétraitement qui vise à lemmatiser le texte en utilisant l'outil MACAON (Nasr *et al.*, 2010) et à supprimer les mots-vides (en anglais *stop word*).
2. Les mots fréquents extraits par TF-IDF vont servir à formuler des requêtes : nous avons pris les 10 mots-clés ayant les scores TF-IDF les plus élevés puis ces mots clés ont été combinés pour effectuer des requêtes constituées chacune de 3 mots.
3. Extraire des documents à partir du web : les requêtes sont soumises à un moteur de recherche sur web (Google) et les pages pointées par les liens retournés sont téléchargées.
4. Nettoyer les données issues du web : les ML ont besoin d'être entraînés sur des corpus propres et normalisés afin de garantir un certain niveau de qualité. Dans notre cas où la source des données d'adaptation est le web, il est important de nettoyer le texte en éliminant les ponctuations, les URLs, les adresses mail et en transformant les formules mathématiques ainsi que les chiffres en des séquences de mots.
5. Construire un modèle de langage du domaine en utilisant les données collectées du web.
6. Adapter le modèle de langage par interpolation linéaire de deux modèles : le modèle générique et le modèle du domaine.

4.2 Résultats

La qualité de l'adaptation des modèles de langage est évaluée à l'aide de la métrique d'évaluation taux d'erreurs mots WER (Pallett, 2003), couramment utilisée dans la littérature pour l'analyse des performances des systèmes de reconnaissance de la parole. Elle se calcule ainsi :

$$WER = \frac{S + I + D}{N} \tag{1}$$

où S est le nombre de mots remplacés par le système , I est le nombre de mots insérés par le système, D est le nombre de mots supprimés par le système, et N est le nombre total de mots dans la référence.

Le tableau 1 présente les premiers résultats expérimentaux. Nous avons adapté le modèle de langage pour les deux cours *Introduction à l'informatique* et *Introduction à l'algorithmique*.

Cours	**% WER sans adaptation**	**% WER avec adaptation**
Introduction à l'informatique	16,2	15,2
Introduction à l'algorithmique	19,6	18,7

TABLE 2 – Résultats d'adaptation du modèle de langage

Les premiers résultats d'adaptation des modèles de langage montrent un gain en terme de WER et sont encourageants pour améliorer la qualité des transcriptions générées pendant les cours.

5 Conclusion

La construction de corpus est une étape cruciale pour tout système de traitement automatique du langage naturel. Nous avons présenté dans cet article un travail portant sur la création d'un corpus annoté, recueilli pour développer une application pour la transcription, la segmentation, l'enrichissement automatique en temps réel et d'autres applications à vocation pédagogique. Ce corpus a été créé dans le cadre du projet PASTEL. Il comprend plus de neuf heures de parole transcrites et annotées. Des expériences préliminaires ont été réalisées pour évaluer notre système de reconnaissance automatique de la parole.

Remerciements

Nous remercions l'agence ANR pour son financement à travers le projet PASTEL sous le numéro de contrat ANR-16-CE33-0007.

Références

AUER E., RUSSEL A., SLOETJES H., WITTENBURG P., SCHREER O., MASNIERI S., SCHNEIDER D. & TSCHÖPEL S. (2010). Elan as flexible annotation framework for sound and image processing detectors. In *Seventh conference on International Language Resources and Evaluation [LREC 2010]*, p. 890–893 : European Language Resources Association (ELRA).

BARRAS C., GEOFFROIS E., WU Z. & LIBERMAN M. (2001). Transcriber : development and use of a tool for assisting speech corpora production. *Speech Communication*, **33**(1-2), 5–22.

BOUCHEKIF A., CHARLET G. D. N. C. D. & ESTÈVE Y. (2015). Segmentation et titrage automatique de journaux télévisés. In *Conférence sur le Traitement Automatique des Langues Naturelles (TALN)*.

CAILLET M., PESSIOT J.-F., AMINI M.-R. & GALLINARI P. (2004). Unsupervised learning with term clustering for thematic segmentation of texts. In *Coupling approaches, coupling media and coupling languages for information retrieval*, p. 648–657 : LE CENTRE DE HAUTES ETUDES INTERNATIONALES D'INFORMATIQUE DOCUMENTAIRE.

GALLEY M., MCKEOWN K. R., FOSLER-LUSSIER E. & JING H. (2003). Discourse segmentation of multi-party conversation. In *Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics*.

GRAVIER G., BONASTRE J., GEOFFROIS E., GALLIANO S., MCTAIT K. & CHOUKRI K. (2004). Ester, une campagne d'évaluation des systèmes d'indexation automatique d'émissions radiophoniques en français. *Proc. Journées d'Etude sur la Parole (JEP)*.

HO I., KIYOHARA H., SUGIMOTO A. & YANA K. (2005). Enhancing global and synchronous distance learning and teaching by using instant transcript and translation. In *Cyberworlds, 2005. International Conference on*, p. 5–pp : IEEE.

HWANG W.-Y., SHADIEV R., KUO T. C. & CHEN N.-S. (2012). Effects of speech-to-text recognition application on learning performance in synchronous cyber classrooms. *Journal of Educational Technology & Society*, **15**(1), 367.

LECORVÉ G., GRAVIER G. & SÉBILLOT P. (2008). An unsupervised web-based topic language model adaptation method. In *Acoustics, Speech and Signal Processing, 2008. ICASSP 2008. IEEE International Conference on*, p. 5081–5084 : IEEE.

NASR A., BÉCHET F. & REY J.-F. (2010). Macaon : Une chaîne linguistique pour le traitement de graphes de mots. In *Traitement Automatique des Langues Naturelles*.

PALLETT D. S. (2003). A look at nist's benchmark asr tests : past, present, and future. In *Automatic Speech Recognition and Understanding, 2003. ASRU'03. 2003 IEEE Workshop on*, p. 483–488 : IEEE.

SHADIEV R., HWANG W.-Y., CHEN N.-S. & YUEH-MIN H. (2014). Review of speech-to-text recognition technology for enhancing learning. *Journal of Educational Technology & Society*, **17**(4), 65.

Apprendre de la littérature scientifique : Les réseaux de signalisation en biologie systémique

Flavie Landomiel[1], Cathy Guérineau[2], Anubhav Gupta[2],
Denis Maurel[2], Anne Poupon[1]
(1) PRC, INRA/CNRS/Université de Tours
(2) Université de Tours, Lifat
Denis.maurel@univ-tours.fr, anne.poupon@inra.fr

RÉSUMÉ

Cet article a pour but de montrer la faisabilité d'un système de fouille de texte pour alimenter un moteur d'inférences capable de construire, à partir de prédicats extraits des articles scientifiques, un réseau de signalisation en biologie systémique. Cette fouille se réalise en deux étapes : la recherche de phrases d'intérêt dans un grand corpus scientifique, puis la construction automatique de prédicats. Ces deux étapes utilisent un système de cascades de transducteurs.

ABSTRACT

Literature-based discovery: Signaling Systems in Systemic Biology.
This paper aims to prove the feasibility of a text mining system to provide an inference engine that builds a signaling system in Systemic Biology. This engine uses predicates extracted from scientific papers. Our text mining system proceeds in two steps. First, it searches interesting sentences in a big scientific corpus. Second, it automatically builds predicates from these sentences. These two steps use finite state transducer cascades.

MOTS-CLÉS : Fouille de texte ; Réseaux de signalisation ; Biologie systémique ; Cascade de transducteur ; Cassys ; Unitex, Prédicats.

KEYWORDS: Text mining; Signaling system; Systemic Biology; Finite state transducer cascades; Cassys; Unitex, Predicates.

1 Introduction

Aujourd'hui, en biologie, la profusion incroyable d'articles scientifiques rend impossible leur suivi par un chercheur. Par exemple, le mot-clé *Signaling* sur Pubmed retourne 480 publications juste pour le mois de septembre 2017. Impossible aussi de "suivre l'actualité" concernant des dizaines de milliers de gènes ou protéines. Des outils automatiques sont nécessaires pour cela.

Le traitement automatique des langues et la fouille de texte sont largement utilisés dans le domaine clinique (Demner-Fushman et Elhadad, 2016 ; Meystre et al., 2008) et biomédical (Zweigenbaum et al., 2007) avec de nombreuses campagnes d'évaluation (Huang et Lu, 2015). Il s'agit pour les chercheurs de retrouver des articles parmi des milliers d'autres. Mais aussi, de découvrir de

nouveaux résultats scientifiques en mettant "bout-à-bout" des résultats disséminés, la *Literature-based discovery* (Weeber et al., 2005). Notre projet vise la détection des interactions entre protéines et/ou gènes. Plusieurs travaux ont déjà été effectués dans ce sens, principalement en étudiant les co-occurrences entre protéines dans les articles scientifiques (Franceschini et al., 2013), la recherche s'effectuant sur des résumés (Hoffmann et Valencia, 2004) ou sur le texte complet (Rzhetsky et al., 2004 ; Miwa et al., 2010 ; Bunescu et al., 2006). Cependant seulement 30% des paires de protéines détectées sont réellement en interaction. Notre projet se rapproche plus des campagnes d'évaluation BioCreative, particulièrement la tâche 5[1] ou BioNLP-ST[2].

Cependant, il en diffère fortement pour la raison suivante : nous avons choisi pour notre part de cibler uniquement la partie factuelle des articles biologiques, à savoir les résultats des expériences réalisées. Ceux-ci, dans un article de biologie ou de médecine, se trouvent dans la partie "Résultats" où sont décrites précisément les expériences. Nous évitons en particulier les introductions et les discussions, tout comme les interprétations des auteurs, ainsi que les citations ou références à des travaux antérieurs. Notre but ultime est la construction automatique de réseaux de signalisation en biologie systémique. Voici notre schéma d'action : créer un corpus d'articles scientifiques à partir de trois mots-clés (deux protéines et un prédicat) ; extraire de ce corpus ce que nous considérons comme des *phrases d'intérêt* ; transformer ces phrases en relations *prédicat-arguments* ; et, enfin, utiliser un système expert pour construire un réseau de signalisation à partir des structures prédicatives mises en évidence. Ensuite, nos résultats seront testés par des collègues biologistes afin de prouver leur véracité scientifique.

La recherche des phrases d'intérêt et leur transformation en structure prédicative sera réalisée en utilisant un système de cascades de transducteurs (Abney, 1996 ; Friburger et Maurel, 2004) implanté dans le logiciel libre Unitex (Paumier, 2003) sous forme de cascades de graphes (Maurel et al., 2011). La construction du réseau de signalisation se fera par un système d'inférence automatique basé sur les travaux de (Gloaguen et al., 2011) et (Rougny et al., 2013). Ce système fonctionnait jusqu'à présent avec des prédicats écrits manuellement par des lecteurs humains et a déjà démontré son efficacité. Notre objectif ici est donc de prouver la faisabilité d'un système de fouille de texte pour alimenter ce moteur d'inférences. Dans l'état actuel du projet, la première partie est réalisée, la seconde dispose d'un premier prototype aux résultats encourageants et la troisième est à venir.

Cet article présentera donc la biologie systémique et les réseaux de signalisation (section 2), puis la recherche des phrases d'intérêt (section 3) et la construction des structures prédicatives (section 4), avant de conclure en présentant la suite du travail.

2 Biologie systémique et réseaux de signalisation

Les cellules développent des réponses spécifiques aux stimuli envoyés par l'organisme, le plus souvent par la mise en circulation d'hormones qui se lient à des récepteurs spécifiques à la surface des cellules. Cette liaison déclenche des cascades de réactions moléculaires, appelées transduction du signal. Nous nous focalisons sur la transduction du signal par les récepteurs couplés aux protéines G (RCPG), qui correspondent à plus de huit cents récepteurs différents. Ce sont des cibles pharmaceutiques idéales qui représentent aujourd'hui environ 40% des médicaments sur le marché. Or, seulement 15% des récepteurs sont « utilisés » par la pharmacopée. Les voies de signalisation

[1] http://www.biocreative.org/tasks/biocreative-vi/track-5/
[2] http://2016.bionlp-st.org/tasks/ge4

sont encore mal connues ; une meilleure connaissance permettra la mise au point de médicaments plus efficaces et ayant moins d'effets secondaires indésirables

La mise au point et les tests qui suivent ont été réalisés pour deux protéines (*ERK* et *arrestin*) et un prédicat (*phosphorylation*). Nous avons donc téléchargé en interrogeant sur ces trois mots-clés tous les articles scientifiques disponibles au format texte sur les bases Istex et NCBI[3]. Soit 3 255 documents. Comme il a été dit en introduction, nous nous focalisons sur la partie "Résultats" et nous avons donc éliminés les textes pour lesquels les trois mots clés utilisés ne se trouvaient pas tous les trois dans cette partie. Ce qui nous a permis de constituer un corpus de 1 282 documents (soit 40% des documents initialement téléchargés).

3 Recherche des phrases d'intérêt[4]

Nous appelons *phrase d'intérêt*, une phrase, de la partie "Résultats", contenant un groupe verbal qui fait référence à la démonstration d'un résultat scientifique impliquant deux protéines minimum. Pour la recherche des phrases d'intérêt, nous avons extrait, de notre corpus de 1 282 documents, un corpus de travail de cinq articles scientifiques et un corpus d'évaluation de vingt-sept articles pris au hasard parmi les 1 282 documents disponibles. Ce corpus représente des articles divers et variés en terme d'année de parution, de nationalité de l'auteur et de journal de publication.

3.1 Étude du corpus de travail

Nous illustrons ci-dessous notre concept de *phrases d'intérêt* par des exemples extraits d'un de nos cinq articles, celui de (Wang et al., 2005) :

1. We found that only phosphorylated ERK bound to Cdc25A.
2. These data provide evidence that the Cdc25A-ERK interaction can be independent of EGFR activation.
3. As shown in Figure 1B, GST-Cdc25A bound to ERK in vitro, whereas glutathione almost completely blocked this binding.

Ces phrases font directement référence à l'article lui-même : soit par un résultat d'expérience (*We found*) ; soit en résumant les diverses démonstrations citées dans le paragraphe *These data provide* ; soit en mentionnant la figure démontrant la phrase conclusive *As shown in Figure 1B*.

A l'inverse, nous voulons éviter de retrouver des phrases parasites. Parmi celles-ci, les phrases descriptives annonçant ce que les auteurs ont l'intention de montrer, par exemple :

4. We next examined whether Cpd 5-induced ERK phosphorylation can be independent of MEK, its direct up-stream kinase activator.

Mais aussi des phrases qui font le lien avec des résultats précédemment publiés ou avec de la bibliographie, telles que :

5. We previously reported that Cpd 5, a Cdc25A inhibitor, caused prolonged EGFR activation, which in turn triggered ERK phosphorylation and cell growth inhibition (Wang et al., 2000, 2002).

[3] https://www.istex.fr/ et https://www.ncbi.nlm.nih.gov/.

[4] Cette partie a fait l'objet d'une communication dans un atelier (Landomiel et al., 2017).

3.2 Les prétraitements

Une des fonctions principales d'Unitex concerne la création et l'application de dictionnaires spécifiques pour l'analyse de corpus. Dans le cas de notre projet, nous avons créé quatre dictionnaires pour prendre en compte la totalité des termes propices à notre analyse : tout d'abord un dictionnaire rassemblant les diverses techniques mises en place en laboratoire, ainsi que tout le lexique inhérent au domaine (issu de la bibliographie ainsi que des sites spécialisés) ; puis un dictionnaire de protéines (issu de la base de données *UniProt*) et deux dictionnaires comprenant les divers systèmes cellulaires et les composés chimiques usuels (issus des sites spécialisés et du NCBI).

Ce dictionnaire est fléchi et complété par deux graphes : le premier pour les formes conjuguées avec des auxiliaires, le second pour certaines formes polylexicales spécifiques aux anticorps. Ces deux graphes sont précédés d'un graphe de découpage en phrases, version anglaise inspirée de (Friburger et al., 2000).

3.3 La cascade

Suite à l'annotation manuelle qui nous a permis de mettre en lumière les constructions de phrases récurrentes dans les articles et la création des dictionnaires, nous avons créé trois autres graphes pour décrire les relations entre les protéines (Figure 1).

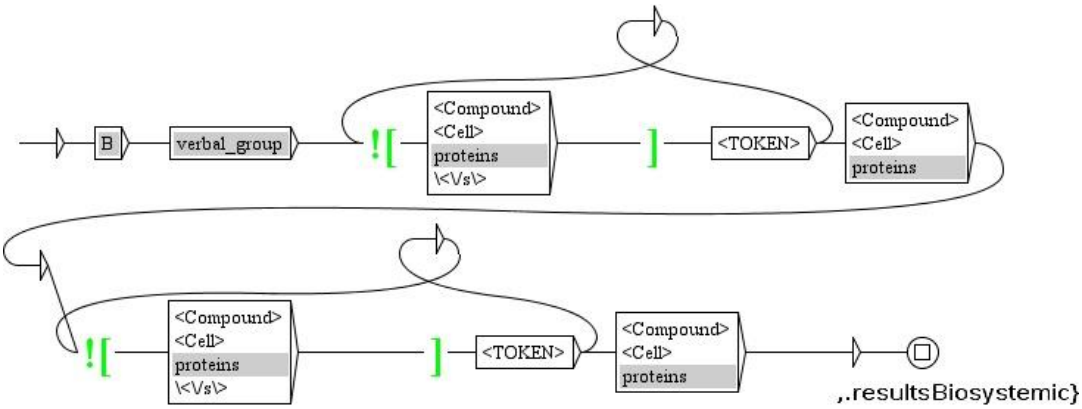


Figure 1 : Un des graphes désignant les relations entre les protéines

Notre cascade est subdivisée en deux sous-cascades. À la fin de la première sous-cascade, toutes les phrases d'intérêt sont identifiées en entier et le texte est balisé au format XML propre à CasSys. Dans la seconde sous-cascade, les balises extra-numéraires sont supprimées ainsi que les phrases qui pourraient être retrouvées par leur structure, mais dont le sens n'est pas recherché. Les verbes trop descriptifs tels que *was collected*, *was probed* mais aussi les phrases commençant par *We next*, *To further* ou encore *In order* qui ne vont pas aboutir à des phrases conclusives y sont intégrés. Enfin, les deux derniers graphes éliminent toutes les balises intermédiaires et crée un fichier rassemblant les phrases clés à la suite les unes des autres.

Cette cascade a été passée sur les 1 282 documents de notre corpus, ce qui nous donne un total de 62 655 phrases extraites.

3.4 Évaluation

Comme il a été dit, notre corpus d'évaluation compte vingt-sept articles, ce qui représente environ 14 000 phrases, dont 4 000 dans la partie résultats. Nous avons calculé les mesures classiques de rappel et de précision (Table 1)

Rappel	90%
Précision	81%

Table 1 : Les résultats de l'évaluation de la première cascade

Malgré des mesures de rappel et de précision très encourageantes, il est toujours possible d'améliorer ces valeurs. Par exemple, nous avons apporté une légère correction au graphe des fins de phrase, en ajoutant la possibilité d'un début de phrase par un chiffre suivi d'une minuscule, ce qui n'avait pas été prévu, mais peut correspondre au nom d'une protéine. D'autre part, les phrases commençant par *when* et celles contenant les mots *previously* et *et al.* avaient été éliminées lors de l'annotation et sont donc non reconnues par la cascade ; or, il s'avère qu'elles correspondent à environ un tiers des phrases manquantes. Il faudra expérimenter si leur réintégration peut augmenter le rappel sans impacter négativement la précision.

4 Construction des structures prédicatives

4.1 Les entités et prédicats traités

Le travail présenté ci-dessous nous a permis de valider notre démarche en testant la construction de trente-cinq entités et prédicats à travers un premier prototype. Quelques exemples sont donnés sur la Table 2. Ces entités et prédicats ont été choisis sur la base des travaux déjà réalisés, cités précédemment, à savoir (Gloaguen et al., 2011) et (Rougny et al., 2013). Ils sont proches de ceux définis dans les campagnes BioCreative et BioNLP-ST évoquées en introduction.

ENTITÉS et PRÉDICAT	DÉFINITION
cell(CL)	CL is a cell type
expressed(G,E,CL,MET)	Gene G is expressed (positive, hypothesis)/not expressed, in Cell type CL, using Method MET
molecule(X)	X is a molecule
particle(X)	X is a particle (molecule, bacteria, virus, etc...)
protein(P)	P is a protein
reactionModulation(X,Y,Z,E,CL,MET)	Signal X modulates the reaction Molecule Y -> Molecule Z, in Cell line CL (can be left void), using Method MET

Table 2 : Exemples de quelques entités et prédicats recherchés

4.2 La cascade

Pour construire les structures prédicatives, une seconde cascade est mise en place. Cette cascade travaille non plus sur les documents du corpus, mais sur les phrases d'intérêt qui en ont été extraites. Elle est, elle aussi, divisée en deux sous-cascades.

La première sous-cascade prépare le document en y repérant les informations qui seront nécessaires à la construction des entités et prédicats. En préliminaire, elle supprime les ellipses concernant les protéines ; ainsi *α- and β-arrestin* devient *α-arrestin and β-arrestin*. Les différentes constructions verbales sont identifiées aussi, l'actif et le passif, mais aussi, c'est important, la négation. Le graphe suivant mets entre balises les métadonnées comme par exemple l'abréviation du mot *Fig* très fréquemment utilisé pour *figure*, mais qui est également le nom d'une protéine. Sans ce graphe, des formules telles que *Fig. 1A* seraient balisées *<Protein>Fig</Protein>. <Gene>1A</Gene>*. Les données numériques résultant des expériences décrites dans les articles sont également balisées afin d'éviter leur intégration dans un nom de gènes. Les graphes suivants repèrent les noms de gènes, protéines ou molécules qui apparaissent dans ces phrases. Puis sont balisées les méthodes utilisées et les prédicats. Voir par exemple la Figure 2.

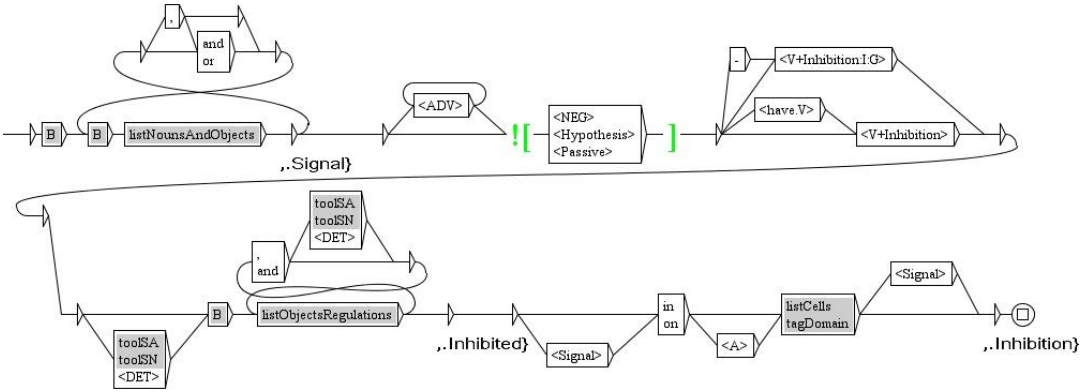


Figure 2 : Le graphe correspondant à un signal d'inhibition

Enfin, la seconde sous-cascade transforme le texte balisé par la première en des structures prédicat-arguments.

La Table 3 présente l'exemple complet d'une séquence issue d'une phrase d'intérêt.

4.3 Évaluation

Pour l'évaluation, nous avons créé un corpus constitué des phrases d'intérêt extraites de cinq documents pris au hasard parmi les 1 282 documents disponibles. Ce corpus représente un total de 226 phrases d'intérêt. Il y a en moyenne 45 phrases d'intérêt par articles.

Comme précédemment, nous avons calculé le rappel et la précision (Table 4). Cependant, les entités étant plus faciles à repérer que les prédicats, nous avons recalculé le rappel et la précision uniquement pour ceux-ci. Le rappel diminue, mais la précision augmente légèrement.

Au final, les résultats sont globalement corrects, sachant que, d'une part, ce prototype sera amélioré et, d'autre part, le système d'inférences qui suit pourra rejeter certains prédicats.

Texte brut	a robust activation of ERKs in δ-OR-expressing HEK-293 cells
Texte balisé	a robust <Activation>activation of <Activated type="Protein">ERKs</Activated> in <Reaction type="Expression"><Cell><Protein>δ-OR</Protein>-expressing <Cell>HEK-293 cells</Cell></Cell></Reaction></Activation>
Entités et prédicats	molecule(ERKs) particle(ERKs) protein(ERKs) cell(HEK-293) molecule(δ-OR) particle(δ-OR) protein(δ-OR) expressed(δ-OR, HEK-293, positive) reactionModulation(unknown signal, ERKs_inactive, ERKs_active, increase, δOR-expressing HEK-293)

Table 3 : Traitement d'une séquence issue d'une phrase d'intérêt

	Entités et prédicats	Prédicats
Rappel	88%	65%
Précision	76%	79%
F-mesure	82%	71%

Table 4 : Les résultats de l'évaluation de la seconde cascade

5 Conclusion et perspectives

Nous venons de montrer la faisabilité d'un système de fouille de texte pour alimenter un moteur d'inférences capable de construire, à partir de prédicats extraits des articles scientifiques, un réseau de signalisation en biologie systémique.

Ceci en deux étapes : la recherche de phrases d'intérêt dans un grand corpus scientifique, puis la construction automatique de prédicats. Ces deux étapes utilisent un système de cascades de transducteurs.

Dans nos perspectives, nous comptons rendre opérationnelle la seconde cascade, afin d'obtenir de meilleurs résultats étendus à une liste plus importantes de prédicats. Puis de créer un enchaînement complet, nos deux cascades et le moteur d'inférence, qui pourra, à partir de la donnée de deux protéines et d'un prédicat, construire des réseaux de signalisation.

Remerciements

Ce projet a été financé par le programme "Chantiers d'avenir" instauré dans le cadre du projet Istex (ANR-10-IDEX-0004-02).

Références

ABNEY S. (1996), Partial Parsing via Finite-State Cascades, *Workshop on Robust Parsing*, 8th *European Summer School in Logic, Language and Information*, Prague, Tchèquie, 8-15.

BUNESCU R., MOONEY R., RAMANI A., MARCOTTE E. (2006). Integrating Co-occurrence Statistics with Information Extraction for Robust Retrieval of Protein Interactions from Medline, in Proceedings of the workshop on linking natural language processing and biology: towards deeper biological literature analysis. 49–56.

DEMNER-FUSHMAN D, ELHADAD N. (2016). Aspiring to Unintended Consequences of Natural Language Processing: A Review of Recent Developments in Clinical and Consumer-Generated Text Processing. Yearb Med. Inform. 224-233.

FRANCESCHINI A., SZKLARCZYK D., FRANKILD S., KUHN M., SIMONOVIC M., ROTH A., LIN J., MINGUEZ P., BORK P., VON MERING C., JENSEN L. J. (2013). STRING v9.1: protein-protein interaction networks, with increased coverage and integration. Nucleic Acids Res. 808-815.

FRIBURGER N., DISTER A., MAUREL D. (2000), Améliorer le découpage des phrases sous Intex, *Revue Informatique et Statistique dans les Sciences Humaines*, vol. 36, n°1-4, p. 181-200.

FRIBURGER N., MAUREL D. (2004), Finite-state transducer cascade to extract named entities in texts, *Theoretical Computer Science*, vol. 313, 94-104.

GLOAGUEN P., CRÉPIEUX P., HEITZLER D., POUPON A., REITER E. (2011). Mapping the follicle-stimulating hormone-induced signaling networks. Front Endocrinol. 2:45

HOFFMANN R., VALENCIA A. (2004). A gene network for navigating the literature. Nature Genetics 36, 664.

HUANG C.C., LU Z. (2015). Community challenges in biomedical text mining over 10 years: success, failure and the future. Brief Bioinform. 17(1):132-44.

LANDOMIEL F., GUPTA A., MAUREL D., POUPON A. (2017). Préliminaire à la construction d'un réseau de signalisation en biologie systémique. Atelier *Fouille de Textes - Text Mine*, en conjonction avec *EGC 2017*, Grenoble, 24 janvier[5].

MAUREL D., FRIBURGER N., ANTOINE J.-Y., ESHKOL-TARAVELLA I., NOUVEL D. (2011). Cascades de transducteurs autour de la reconnaissance des entités nommées. *Traitement automatique des langues*, 52(1):69-96.

MEYSTRE S. M., SAVOVA G. K., KIPPER-SCHULER K. C., HURDLE J. F. (2008). Extracting information from textual documents in the electronic health record: a review of recent research. Yearb Med. Inform. 128-144.

[5] http://vincentlemaire-labs.fr/TM2017/Actes_TextMine17.pdf

MIWA M., SÆTRE R., KIM J.-D., TSUJII J. (2010). Event extraction with complex event classification using rich features. J. Bioinform. Comput. Biol. 08, 131–146.

PAUMIER S. (2003), *De la Reconnaissance de Formes Linguistiques à l'Analyse Syntaxique*, Thèse de Doctorat en Informatique, Université de Marne-la-Vallée.

ROUGNY A., FROIDEVAUX C., YAMAMOTO Y., INOUE K. (2013). Translating the SBGN-AF language into logic to analyze signalling networks. LNMR 2013 (First International Workshop on Learning and Nonmonotonic Reasoning). 44-55.

RZHETSKY A., IOSSIFOV I., KOIKE T., KRAUTHAMMER M., KRA P., MORRIS M., YU H., DUBOUÉ P. A., WENG W., WILBUR W. J., HATZIVASSILOGLOU V., FRIEDMAN C. (2004). GeneWays: a system for extracting, analyzing, visualizing, and integrating molecular pathway data. J. Biomed. Inform. 37, 43–53.

WANG Z., ZHANG B., WANG M., CARR B. I. (2005). Cdc25A and ERK interaction: EGFR independent ERK activation by a protein phosphatase Cdc25A inhibitor, Compound 5. *Journal of Cellular Physiology* 204(2), 437–444.

WEEBER M., KORS J. A., MONS B. (2005). Online tools to support literature-based discovery in the life sciences. Brief. Bioinform. 6, 277–286.

ZWEIGENBAUM P., DEMNER-FUSHMAN D., YU H., COHEN K. B. (2007). Frontiers of biomedical text mining: current progress. Brief Bioinform. 8(5):358-75.

Détection des couples de termes translittérés à partir d'un corpus parallèle anglais-arabe

Wafa Neifar[1,2] Thierry Hamon[1,3] Pierre Zeigenbaum[1]
Mariem Ellouze Khemakhem[2] Lamia Hadrich Belguith[2]
(1) LIMSI, CNRS, Université Paris-Saclay, F-91405 Orsay, France
(2) MIRACL Laboratory, Sfax University, B.P-3018 Sfax, Tunisie
(3) Université Paris 13, Sorbonne Paris Cité, F-93430, Villetaneuse, France
neifar,hamon,pz@limsi.fr, mariem.ellouze@planet.tn, l.Belguith@fsegs.rnu.tn

Résumé

Nous présentons une méthode pour extraire des couples de termes médicaux translittérés de l'anglais en caractères arabes. Nous avons proposé un processus de construction des translittérations de termes anglais en arabe. Celui-ci s'appuie sur une étude en corpus pour la création d'une table de correspondances des caractères anglais en arabe mais aussi sur des règles de conversion qui tiennent compte de certaines particularités de la langue arabe comme l'agglutination et la non-voyellation. Nous avons évalué l'apport de l'utilisation de la translittération pour identifier des couples de termes anglais-arabe sur un corpus parallèle de textes médicaux. Les résultats montrent que parmi 137 couples de mots anglais-arabe extraits, 120 sont jugés corrects (soit 87,59%), dont 107 représentent des couples de termes médicaux (soit 89,16% des translittérations correctes et 78,10% des résultats).

Abstract

Detection of transliterated pairs of terms from an English-Arabic parallel corpus

We aim at extracting pairs of medical terms which are transliterated from English to Arabic characters. We propose a method for building transliteration of English terms in Arabic. This method relies on the study of a corpus in order to define a conversion table but also on rules which take into account specificities of Arabic language such as the agglutination and the absence of diacritisation. We have evaluate the use of this transliteration method for identify pairs of English-Arabic terms in a medical parallel corpus. The results show that among the 137 extracted English-Arabic word pairs, 120 are considered as correct transliterations (87.59%), among which 107 pairs represent medical terms (this represents 89.16% of the correct transliterations and 78.10% of the results).

Mots-clés : Extraction terminologique bilingue, alignement de mots, translittération, corpus parallèle..

Keywords : Bilingual terminology extraction, word alignment, transliteration, parallel corpora..

1 Introduction

Bien que l'arabe standard moderne (MSA) soit la langue officielle de 26 pays, son usage est variable dans les domaines de spécialité. Ainsi, alors qu'en médecine, la langue utilisée lors de la pratique et de l'enseignement est la langue française ou anglaise (Samy *et al.*, 2012), l'arabe est également

une langue officielle de l'OMS (Organisation Mondiale de la Santé) ou d'autres organismes internationaux. De même, l'arabe est la langue de travail dans d'autres domaines comme l'agriculture, la géologie, la protection de l'environnement ou le droit (Massoud, 2003). De nombreux documents administratifs et techniques sont donc rédigés dans cette langue et il est important de disposer de systèmes de gestion terminologique et de procéder à un aménagement terminologique dans de nombreux autres domaines. L'étude des terminologies scientifiques et techniques en arabe montre que celles-ci sont imprégnées des terminologies établies en anglais ou en français, alors considérées comme des terminologies de référence (Lelubre, 2008). Ce constat n'est toutefois pas valable pour tous les domaines. Ainsi, certains termes médicaux anglais sont issus de l'arabe (Wulff, 2004).

Partant de ce constat mais aussi d'observations réalisées en corpus, nous avons choisi d'exploiter la présence de termes anglais dans les terminologies et les textes de spécialité rédigés en arabe. La méthode que nous proposons vise à exploiter la translittération des termes anglais en caractères arabes et le principe du transfert translingue (McDonald *et al.*, 2011). À notre connaissance, il n'existe pas de systèmes de translittération de l'anglais vers l'arabe. L'application de la méthode sur un corpus parallèle anglais-arabe aligné au niveau des mots permet d'extraire des termes arabes après reconnaissance ou extraction des termes anglais.

Afin d'évaluer l'apport de la translittération de termes anglais en caractères arabes sur la construction d'une terminologie bilingue à partir d'un corpus parallèle anglais-arabe de textes médicaux, nous présentons dans cet article un système de translittération qui permet d'extraire des couples de termes médicaux translittérés de l'anglais vers l'arabe. Après un état de l'art des travaux des systèmes de translittération (section 2), nous présentons le corpus que nous utilisons, sa constitution ainsi que son alignement au niveau des mots dans la section 3. Nous décrivons la méthodologie suivie lors de la construction de notre système de translittération (section 4). Nous présentons et discutons les résultats obtenus (section 5) avant de conclure et de fournir quelques perspectives à ce travail (section 6).

2 État de l'art

Plusieurs travaux de recherche ont été menés ces dernières années sur la translittération concernant l'arabe. La plupart des travaux se sont focalisés sur le passage de l'arabe vers l'écriture latine (Al-Onaizan & Knight, 2002 ; Sherif & Kondrak, 2007 ; Saadane & Semmar, 2012).

Ainsi, Semmar & Saadane (2014) proposent un système de translittération des noms propres de l'écriture arabe vers l'écriture latine afin d'améliorer leur processus d'alignement des mots simples et composés. Les équivalences graphémiques utilisées sont établies à partir d'une étude de corpus de textes parallèles français-arabe. La première étape consiste à supprimer les voyelles avant de translittérer le nom. Des règles contextuelles définies pour rendre compte le plus précisément possible des formes observées sont appliquées selon le nombre de consonnes du nom considéré dans un ordre de priorité déterminé. La liste des noms en écriture latine est ensuite normalisée et une pondération est définie comme le nombre d'occurrences retourné par le moteur de recherche Google. Ce système produit en sortie une liste triée de noms arabes écrits en caractères latins. L'évaluation est effectuée sur 283 phrases du corpus du Monde Diplomatique français-arabe. Les taux de précision, rappel et F-mesure du processus d'alignement augmentent respectivement de 85% , 80% et 82% à 88%, 85% et 86%.

Mubarak & Abdelali (2016) présentent une nouvelle approche pour la construction d'un système de translittération. Leur corpus est composé de 881 310 paires de noms de personnes de l'arabe vers l'anglais collectées à partir de Twitter. Les données sont nettoyées pour extraire le nom complet écrit

en arabe ayant un chevauchement au dessus d'un certain seuil, le nom d'utilisateur écrit en caractères latins (sa translittération anglaise), ainsi que le pays de l'utilisateur. Les noms écrits en arabe et en latin et la localisation des utilisateurs sont normalisés selon la méthode décrite dans (Darwish *et al.*, 2012). Ensuite, les marques typographiques et les titres comme Mr, Mme,.. sont supprimés. Les lettres arabes qui n'ont pas de correspondance phonétique exacte dans les langues latines seront remplacées par des chiffres. Les variantes multiples de translittération des caractères arabes en leurs équivalents latins sont extraites. Les noms arabes sont translittérés différemment selon les variations dialectales régionales. Pour mesurer et quantifier la similitude entre les noms en arabe et ceux en latin, le score de similarité est calculé en utilisant la distance d'édition de Levenshtein. Les scores obtenus sur 1000 paires de noms sélectionnées au hasard sont importants : 0,96 pour la précision, 0,97 pour le rappel et de nomss'agit 0,965 pour la F1-mesure.

Les travaux présentés ci-dessus s'intéressent à la translittération des noms propres arabe en anglais, la langue arabe étant considérée comme la langue source. Si notre objectif comporte des similarités avec ces travaux, il diffère sur deux points : nous nous intéressons ici à la translittération inverse (anglais vers arabe), ainsi qu'aux mots, plus particulièrement aux termes de manière générale.

3 Corpus

3.1 Constitution du corpus médical

Comme indiqué dans (Al-Sulaiti & Atwell, 2006), très peu de corpus de spécialité sont disponibles pour l'arabe. C'est particulièrement le cas lorsqu'il s'agit de corpus parallèles. Pour cela, nous avons constitué notre corpus à partir de 133 documents produits par la *National Library of Medicine* (NLM) et disponibles en ligne sur MedlinePlus[1]. Ces documents sont des brochures de quelques pages à destination des patients. Ces brochures apportent des informations sur des problèmes médicaux, décrivent les conditions de réalisation d'examen, fournissent des conseils de comportement face à une maladie, ou pour l'amélioration du bien-être. L'anglais est la langue source, et ces documents sont traduits dans de nombreuses langues cibles comme le français ou l'arabe. Pour la définition de notre approche, nous avons utilisé 71 brochures. Les corpus anglais et arabe contiennent respectivement 35 521 et 33 402 mots. Pour évaluer notre système de translittération, nous avons ajouté 62 documents, soit 25 379 mots pour le corpus anglais et 21 950 mots pour le corpus arabe. Contrairement aux documents en anglais et en français, la conversion au format texte des documents en arabe, en vue de réaliser des traitements automatiques, pose des nombreux problèmes (Habash, 2010) que nous avons tenté de résoudre mais qui rendent la tâche de nettoyage très coûteuse en temps : erreur de forme des caractères, utilisation d'un caractère persan ressemblant graphiquement à un caractère arabe, etc. L'ensemble des documents du corpus est ensuite segmenté par MADAMIRA (Pasha *et al.*, 2014) et aligné au niveau des paragraphes et au niveau des phrases. La qualité de l'alignement a été vérifiée manuellement.

3.2 Alignement au niveau des mots

Un alignement au niveau des mots a été obtenu en utilisant l'outil GIZA++ (Och & Ney, 2003). Nous avons défini trois alignements entre le corpus anglais et le corpus arabe qui prennent en compte diffé-

[1] `http://www.nlm.nih.gov/medlineplus/languages/all_healthtopics.html`

rentes informations morphologiques fournies par MADAMIRA sur le corpus arabe : (Alignement1) l'alignement est réalisé sans traitement morphologique particulier ; il s'agit de notre base de comparaison ; (Alignement2) avant l'alignement, les enclitiques et proclitiques sont désagglutinés du mot arabe auquel ils se rapportent ; (Alignement3) avant alignement, les enclitiques, les proclitiques et les articles sont désagglutinés du mot arabe auquel ils se rapportent. Après l'alignement 3, le corpus arabe comporte 51 951 mots, et la partie ajoutée pour l'évaluation, 27 952 mots.

4 Extraction de paires de termes par translittération

L'un des facteurs les plus importants qui ont contribué à la modernisation rapide de la langue arabe a été l'assimilation du vocabulaire d'origine étrangère. Ainsi, nous observons dans notre corpus que certains termes médicaux arabes sont le produit de la transcription des termes anglais. Pour extraire ces couples de termes anglais-arabe, nous nous sommes basés sur la translittération complète fixée par la norme ISO 233-1 (translittération des caractères arabes en caractères latins) pour créer la correspondance inverse (translittération des caractères latins en caractères arabes). Nous avons ainsi créé notre propre table de correspondance des caractères anglais vers l'arabe en nous appuyant également sur les observations faites sur notre corpus. Celle-ci est ainsi adaptée aux caractéristiques phonologiques utilisées lors du passage de l'anglais à la langue arabe. Nous nous sommes inspirés aussi de la translittération complète fixée par la norme ISO 233-1 (translittération des caractères arabes en caractères latins), sauf que nous faisons le passage à l'inverse. A notre connaissance, il n'existe pas de table de correspondance des caractères latin en arabe.

La translittération vers l'arabe peut être ambiguë. Pour chaque caractère anglais, nous essayons d'identifier les différents caractères arabes qui peuvent lui correspondre. Par exemple, le caractère *g* peut être translitteré par le caractère ج comme dans le couple ***g****lucophage* (جلوكوفاج) ou par le caractère غ comme dans le couple ***g****luconate* (غلوكونات). La table 1 montre un extrait de notre table de correspondance des caractères. Certains couples de mots ont été directement récupérés à partir de notre table de correspondance, alors que d'autres ont nécessité l'enrichissement de notre processus d'extraction par certains traitements complémentaires.

Caractère anglais	Caractères arabes
g	غ ج
r	ر
c	ق ك س

TABLE 1 – Un extrait de la table de correspondance des caractères

Notre méthode d'extraction de couples de termes anglais-arabe par translittération tient compte de certaines particularités de la langue arabe. Comme nous l'avons déjà signalé dans (section 3.2), l'étape de l'alignement repose sur le corpus arabe désagglutiné au niveau des enclitiques, des proclitiques et des articles. Cependant, lorsqu'il s'agit d'un terme emprunté d'une langue étrangère, l'étiqueteur morpho-syntaxique MADAMIRA ne parvient pas à identifier ou à séparer les articles et les proclitiques du mot en question. Ainsi, la présence de ces particules provoque des erreurs lors de l'extraction des couples de termes anglais-arabe transcrits. La prise en compte de leur présence éventuelle a nécessité l'ajout de certains traitements supplémentaires au sein de notre processus d'extraction. Il s'agit d'identifier les articles et les propositions situés au début des termes arabes pour avoir une

meilleure correspondance de couple de termes anglais-arabe transcrits. La table 2 présente quelques exemples de couples extraits grâce à cette étape.

TABLE 2 – Amélioration de l'extraction suite au traitement de l'agglutination

Terme anglais	Terme arabe aggluti né	Terme arabe traité
titanium	التيتانيوم *(et le titanium)*	تيتانيوم *(titanium)*
progesterone	والبروجسترون *(et le progestérone)*	بروجسترون *(progestérone)*
steroid	بالاستيرويد *(par le stéroïde)*	ستيرويد *(stéroïde)*

Outre la prise en compte de l'agglutination telle que décrite précédemment, nous tenons compte de la non-voyellation. En effet, comme la plupart des textes en MSA, notre corpus arabe n'est pas voyellé[2]. Il est alors possible d'observer différentes formes de transcription du même terme anglais qui tiennent compte ou non de la présence de voyelles longues. Lors du passage des termes anglais à la langue arabe par transcription, les terminologues tentent de remplacer les voyelles latines par des voyelles arabes courtes, alors que d'autres mettent l'accent sur ces voyelles en les remplaçant par des voyelles arabes longues. Par exemple, le terme *oxygen* (*oxygène*) est transcrit trois fois sous la forme اوكسجين (`oksjin`) où la voyelle *o* est représentée par le caractère ا suivie par la voyelle longue و et six fois sous la forme اكسجين (`owksjin`) où la voyelle longue 'و' n'est pas représentée.

Même si les termes arabes translittérés proviennent d'autres langues étrangères, ceux-ci suivent les règles imposées par la langue arabe pour la production des formes du pluriel des noms et des adjectifs. Nous normalisons donc les termes anglais et arabes en les mettant au singulier. Par exemple, le passage du pluriel au singulier nous permet d'identifier le couple de termes ***protein*** - بروتين *(protéine)* dont le terme arabe apparait dans le corpus sous la forme du pluriel بروتينات *(des protéines)*. De même, pour le couple ***hormone*** - هورمون *(hormone)*, chacun des termes apparaît dans le corpus sous la forme du pluriel ; *hormones* pour la partie anglaise et هرمونات pour celle en arabe *(des hormones)*.

Pour calculer la correspondance entre un mot anglais et un mot arabe, nous commençons par translittérer le mot anglais à l'aide de notre table. Cette première étape tient compte des différentes correspondances pour chaque caractère. Plusieurs traitements sont également appliqués au mot arabe (désagglutination au niveau des proclitiques et des articles, passage de la forme du pluriel au singulier) ainsi qu'à notre translittération (comme l'ajout ou suppression des voyelles longues). Par exemple, le terme anglais **bacteria** (*(bactérie)*) est translittéré en arabe soit sous la forme بكتريا [`bktria`], soit sous la forme بكتيريا [`bktiria`] en tenant compte de la voyelle longue. Dans la deuxième forme arabe, la voyelle 'e' de 'bacteria' est remplacée par la voyelle arabe longue 'ي'[3]

L'ensemble des étapes effectuées sert à tester si notre translittération peut correspondre, tout en appliquant certains traitements, au mot arabe en question. Si c'est le cas, nous considérons que le couple de termes anglais-arabe représente une translittération du terme anglais en caractères arabes.

[2]L'arabe standard moderne se caractérise par la présence de symboles optionnels non alphabétiques, appelés diacritiques. En leur absence, la connaissance de la langue permet de prononcer correctement le mot. Dans certains cas, l'absence de voyellation peut donc engendrer des ambiguïtés de compréhension. Ce constat est d'autant plus vrai lorsque les textes arabes non-voyellés sont analysés automatiquement.

[3]La différence graphique est liée au changement des allographes des caractères arabes selon leur position dans un mot ou leur position indépendante (lorsque le caractère est représenté seul).

5 Expériences et résultats

Nous avons évalué l'impact de l'analyse morphologique dans la qualité de l'alignement. Puis, nous évaluons la qualité de la translittération et l'utilisation des résulats obtenus pour détecter des couples de termes anglais-arabe. Puis nous présentons une analyse des erreurs.

Prise en compte de l'analyse morphologique lors de l'alignement L'alignement au niveau des mots est une étape préliminaire à notre méthode d'extraction des couples de termes anglais-arabe. Pour l'évaluer, nous avons considéré que le taux d'estimation de correspondance fourni par Giza++ reflète la qualité de l'alignement. Le taux de correspondance de l'alignement initial (pas d'analyse morphologique en préalable à l'alignement) est estimé à 75,98%. Nous observons une amélioration des taux de correspondance lorsque les enclitiques et les proclitiques sont désagglutinés (79,41%, +3,43) et encore après désagglutination des articles (82,45%, +3,07).

Translittération et extraction des couples de termes anglais-arabe Nous avons utilisé les 133 documents à notre disposition afin d'avoir une bonne qualité d'alignement. Le système de translittération extrait 137 couples de termes anglais-arabe. Nous les avons évalué manuellement. Ainsi, 120 couples de mots sont jugés corrects (soit 87,59%). Parmi ceux-ci, 107 couples sont des couples de termes médicaux (soit 89,16% des translittérations correctes et 78,10% des résultats).

Nous avons observé que 7,29% des couples sont bien alignés mais ils ne représentent pas une translittération. Par exemple, le pronom relatif anglais *that* (*qui*) est associé au mot arabe التي (`alaty`, pronom relatif féminin singulier). Même si la translittération de certains d'entre eux donne un mot arabe qui en est la traduction, ceci n'est pas le résultat d'une écriture en caractères arabes des mots anglais sources. C'est le cas du mot *african* (*africain*) dont le mot arabe est افريقي(`afriqi`). 5,1% des couples de mots extraits ne sont ni alignés ni translittérés. Par exemple, le mot *grounds* (*marc*) est aligné avec un point ".", signe de ponctuation, ou aussi le mot *your* (*ton/ta/tes/votre/vos*) qui est aligné avec l'article arabe ال (*le/la/les*).

À partir des résultats obtenus, nous constatons que la terminologie arabe contient certains termes anglais translittérés même s'il existe déjà un terme arabe qui lui correspond. Autrement dit, un terme anglais peut à la fois avoir comme correspondant sa translittération en caractères arabes ainsi qu'un autre terme arabe ayant le même sens. Par exemple, le terme *ounces* (*onces*) peut correspondre au terme arabe اوقيات ou à sa translittération donnant le terme arabe اونصة.

Analyse des erreurs Comme il s'agit d'un alignement au niveau des mots, certains termes complexes sont alignés avec une partie du terme arabe qui serait le correspondant complet du terme anglais en question. Dans l'exemple suivant, nous constatons que le terme anglais *x-ray* (*rayon-X*) est aligné avec le terme simple arabe اشعة (*rayons*) qui représente une partie du terme complexe اشعة اكس. Dans le corpus, le tiret qui lie les deux parties du terme arabe n'existe pas. À cause de cela, l'alignement est fait avec une seule partie du terme. L'utilisation des acronymes est un phénomène fréquent lors de la construction de la terminologie médicale anglaise. Celle-ci influe sur les résultats de translittération obtenus. Par exemple, L'acronyme *(MRSA)* (*SARM : Staphylococcus aureus résistant à la méthicilline*) est préféré à la forme longue du terme *Methicillin-Resistant Staphylococcus Aureus*.

L'acronyme est alors aligné avec le mot arabe للمثيسيلين[4] (*à la méticilline*). Le terme arabe extrait représente une partie du terme arabe complexe العنقوديات الذهبية المقاومة للمثيسيلين qui lui correspond. Une autre caractéristique de la terminologie anglaise est l'utilisation des abréviations. Ainsi, des couples de termes dont la partie arabe représente une translittération ne sont pas extraits par notre système. Par exemple, le couple des termes *flu* et انفلونزا (*grippe/influenza*) n'est pas retrouvé car le terme arabe correspond à la translittèration du mot complet anglais *Influenza* et non à son abréviation *flu*.

Certains termes arabes translittérés ne figurent pas dans la liste produite par notre système suite aux erreurs produites lors de la phase d'alignement. Par exemple, le terme arabe زنك (*zinc*) est aligné avec le terme *oxide* alors que اكسيد (*oxide*) est aligné avec le terme *zinc*.

Comme nous l'avons déjà signalé, la construction de la terminologie médicale arabe repose sur l'assimilation du vocabulaire d'origine étrangère. L'origine des termes translittérés influe sur les résultats obtenus. Les résultats dépendent donc des langues utilisées dans le corpus. Si notre corpus était français-arabe, il aurait existé d'autres couples de translittérations. Par exemple, le terme arabe بوصة est une translittération du terme français *pouce*. Pour cette raison, le couple de termes anglais-arabe (*inch*, بوصة) ne figure pas dans la liste produite par notre système de translittération.

6 Conclusion et perspectives

Nous nous sommes intéressés à la construction d'un système de translittération afin d'extraire les couples de termes anglais-arabe translittérés de l'anglais en caractères arabes. Notre objectif est d'évaluer l'apport de la translittération dans la construction d'une terminologie bilingue à partir d'un corpus médical parallèle anglais-arabe. Nous avons proposé un processus de détection des translittérations de termes anglais en arabe. Pour cela, nous nous sommes appuyés sur une étude du corpus pour la création d'une table de correspondances des caractères anglais en arabe mais nous avons également pris en compte certaines particularités de la langue arabe comme les phénomènes d'agglutination et de non-voyellation. Nous avons ainsi défini des traitements supplémentaires qui tiennent compte de ces phénomènes. Les expériences réalisées sur un corpus anglais de 55 352 mots et un corpus arabe de 60 900 mots ont permis d'obtenir 87,59% de translittérations correctes. Parmi celles-ci, 78,10% représentent des couples de termes médicaux translittérés de l'anglais en caractères arabes.

Plusieurs perspectives de travail s'offrent à nous. D'une part, le processus d'alignement des textes au niveau des mots doit être amélioré . D'autre part, nous envisageons d'élargir notre corpus de test. Enfin, en fonction de la disponibilité de corpus parallèles en MSA, nous proposons d'évaluer notre travail dans d'autres domaines de spécialité.

Remerciements

Ce travail a bénéficié d'un soutien de l'action Horizon 2020 Marie Sklodowska-Curie Innovative Training Networks — European Joint doctorate (ITN-EJD) de l'Union européenne, projet N° 676207 (MiRoR) et de l'ANR dans le cadre du projet CLEAR (ANR-17-CE19-0016-01).

[4]Ce terme arabe n'est pas désagglutiné car il s'agit d'un mot étranger que MADAMIRA n'arrive pas à identifier.

Références

Al-Onaizan Y. & Knight K. (2002). Translating named entities using monolingual and bilingual resources. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, ACL '02, p. 400–408.

Al-Sulaiti L. & Atwell E. (2006). The design of a corpus of Contemporary Arabic. *International Journal of Corpus Linguistics*, **11**(1), 1–36.

Darwish K., Magdy W. & Mourad A. (2012). Language processing for arabic microblog retrieval. In *Proceedings of the 21st ACM International Conference on Information and Knowledge Management (CIKM '12)*, p. 2427–2430 : ACM.

Habash N. (2010). *Introduction to Arabic Natural Language Processing*. Synthesis Lectures on Human Language Technologies. Morgan & Claypool Publishers.

Lelubre X. (2008). La constitution de la terminologie arabe de la physique : aspects diachroniques. *Travaux du CRTT*. Article dans numéro spécial Travaux du CRTT.

Massoud R. (2003). La terminologie au liban : réalités et défis. *Annales de l'Institut de langues et de traduction (ILT)*, **10**.

McDonald R., Petrov S. & Hall K. (2011). Multi-source transfer of delexicalized dependency parsers. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, EMNLP '11, p. 62–72.

Mubarak H. & Abdelali A. (2016). Arabic to english person name transliteration using twitter. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*.

Och F. J. & Ney H. (2003). A systematic comparison of various statistical alignment models. *Computational Linguistics*, **29**(1), 19–51.

Pasha A., Al-Badrashiny M., Diab M., Kholy A. E., Eskander R., Habash N., Pooleery M., Rambow O. & Roth R. (2014). Madamira : A fast, comprehensive tool for morphological analysis and disambiguation of arabic. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, Reykjavik, Iceland : European Language Resources Association (ELRA).

Saadane H. & Semmar N. (2012). Utilisation de la translittération arabe pour l'amélioration de l'alignement de mots à partir de corpus parallèles français-arabe. In *Proceedings of the Joint Conference JEP-TALN-RECITAL 2012, volume 2 : TALN*, p. 127–140 : ATALA/AFCP.

Samy D., Moreno-Sandoval A., Bueno-Díaz C., Garrote-Salazar M. & Guirao J. M. (2012). Medical term extraction in an arabic medical corpus. In *Proceedings of LREC'12*, p. 640–645.

Semmar N. & Saadane H. (2014). Etude de l'impact de la translittération de noms propres sur la qualité de l'alignement de mots à partir de corpus parallèles français-arabe). In *Traitement Automatique des Langues Naturelles (TALN 2014)*, p. 268–279, Marseille, France.

Sherif T. & Kondrak G. (2007). Bootstrapping a stochastic transducer for arabic-english transliteration extraction. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, p. 864–871, Prague, Czech Republic : Association for Computational Linguistics.

Wulff H. (2004). The language of medicine. *Journal of the Royal Society of Medicine*, **97**, 187–188.

Utilisation d'une base de connaissances de spécialité et de sens commun pour la simplification de comptes-rendus radiologiques

Lionel Ramadier[1] Mathieu Lafourcade[2]
(1) LIMSI, Campus universitaire bât 508 Rue John von Neumann, 91405 Orsay, France
(2) LIRMM, 161 rue Ada, 34095 Montpellier, France
ramadier@limsi.fr, mathieu.lafourcade@lirmm.fr

RÉSUMÉ

Dans le domaine médical, la simplification des textes est à la fois une tâche souhaitable pour les patients et scientifiquement stimulante pour le domaine du traitement automatique du langage naturel. En effet, les comptes rendus médicaux peuvent être difficile à comprendre pour les non spécialistes, essentiellement à cause de termes médicaux spécifiques (*prurit*, par exemple). La substitution de ces termes par des mots du langage courant peut aider le patient à une meilleure compréhension. Dans cet article, nous présentons une méthode de simplification dans le domaine médical (en français) basée sur un réseau lexico-sémantique. Nous traitons cette difficulté sémantique par le remplacement du terme médical difficile par un synonyme ou terme qui lui est lié sémantiquement à l'aide d'un réseau lexico-sémantique français. Nous présentons dans ce papier, une telle méthode ainsi que son évaluation.

ABSTRACT

Radiological text simplification using a general knowledge base.

In the medical domain, text simplification is both a desirable and a challenging natural language processing task. Indeed, first, medical texts can be difficult to understand for patient, because of the presence of specialized medical terms. Replacing these difficult terms with easier words can lead to improve patient's understanding. In this paper, we present a lexical network based method to simplify health information in French language. We deal with semantic difficulty by replacement difficult term with supposedly easier synonyms or by using semantically related term with the help of a French lexical semantic network. We extract semantic and lexical information present in the network. In this paper, we present such a method for text simplification along with its qualitative evaluation. .

MOTS-CLÉS : TALN, médical, simplification, réseau lexico-sémantique.

KEYWORDS: NLP, BioNLP, text simplification, lexico-semantic network.

1 Introduction

La simplification de texte (ST) est un domaine du traitement automatique du langage naturel (TALN) dont le but est de rendre des textes plus compréhensibles tout en garantissant l'intégrité de leur contenu et de leur structure. Dès lors, la ST peut être un moyen d'aider des personnes à accéder à la compréhension de documents écrits spécialisés. En effet, les problèmes de compréhension sont souvent dus à une grande complexité des textes, tant au niveau lexical que syntaxique. Dans ce cadre, la ST peut être vue comme une tâche de traduction mono-langue, où la langue source a besoin d'être

traduite en une version simplifiée de la même langue.

Son application dans le domaine médical revêt une importance particulière. La compréhension d'un texte médical peut être particulièrement ardue pour les patients non spécialistes du domaine médical. Les textes médicaux sont difficiles à comprendre pour un non expert (Keselman & Smith, 2012) du fait que les médecins écrivent souvent avec des termes spécialisés (*ataxie*) et des abréviations (*SA* pour *sans aménorrhée*) qui nécessitent une certaine connaissance médicale. (Chapman *et al.*, 2003) et (Lerner *et al.*, 2000) ont déjà montré que les termes médicaux pouvaient être un obstacle à la compréhension pour les patients. Ces difficultés peuvent avoir un impact négatif sur la communication entre les patients et les médecins, et les soins offerts aux malades (Tran *et al.*, 2009).

Cependant, il existe peu de travaux sur des méthodes automatiques de simplification des textes médicaux et plus particulièrement des comptes rendus cliniques (Keselman *et al.*, 2008). Le but de notre travail est de simplifier des comptes rendus radiologiques en français grâce à une base de connaissance de sens commun qui contient à la fois de la connaissance générale mais aussi de spécialité. Notre approche concerne la simplification lexicale. Nous utilisons pour cela non seulement des synonymes mais aussi des termes liés par des relations hiérarchiques et/ou sémantiques. Nous présentons dans un premier temps les travaux liés à l'état de l'art (section 2). Nous présentons ensuite notre approche de simplification utilisée (section 3). Nous décrivons et discutons les résultats obtenus (sections 4 et 5). Nous concluons avec des perspectives pour les travaux futurs.

2 Etat de l'art

Le niveau de difficulté peut varier entre différents types de textes médicaux (Leroy *et al.*, 2010) et même les brochures pour patient peuvent être difficile à comprendre (Kokkinakis *et al.*, 2012). La simplification lexicale aide à rendre un texte plus compréhensible. En effet, (Abrahamsson *et al.*, 2014) ont montré que le remplacement des mots difficiles par des synonymes du langage courant pouvait réduire le niveau de difficulté d'un texte. La substitution par les synonymes a été évaluée sur des textes médicaux anglais (Leroy *et al.*, 2012) (Slaughter *et al.*, 2005) mais aussi suédois (Keskisärkkä, 2012). (Leroy *et al.*, 2012) utilise un lexique de synonymes et remplace les termes difficiles avec des synonymes du langage courant. Le niveau de difficulté d'un mot est déterminé par sa fréquence d'apparition dans un corpus général. Dans (Zeng-Treitler *et al.*, 2007), les auteurs utilisent deux stratégies pour réduire la difficulté lexicale des textes médicaux (le remplacement par des synonymes et par l'ajout d'explication). Pour effectuer la simplification des textes à l'aide de synonymes par exemple, des ressources spécifiques sont nécessaires. Dans le domaine médical, ces ressources se présentent souvent sous forme de lexiques où les termes sont mis en correspondance avec les expressions non spécialisées correspondantes. L'utilisation de ce type de lexique est apparue avec le travail collaboratif Consumer Health Vocabulary (CHV) (Zeng-Treitler *et al.*, 2007) (Qenam *et al.*, 2017). (Elhadad & Sutaria, 2007) a construit un lexique de termes alignés avec leurs équivalents non techniques à partir de l'UMLS (Unified Medical Language System [1]). (Leroy *et al.*, 2012) ont développé un système qui utilise la familiarité d'un terme pour identifier la difficulté du texte et sélectionne des termes plus compréhensibles à partir de ressources lexicales comme WordNet [2], UMLS et Wiktionnary [3]. Dans le domaine de la radiologie, plusieurs études ont montré que les comptes rendus radiologiques sont parmi les plus difficiles à comprendre (Keselman *et al.*, 2007). Une

1. https ://www.nlm.nih.gov/research/umls/
2. https ://wordnet.princeton.edu/
3. https ://www.wiktionary.org/

équipe suédoise (Kvist & Velupillai, 2013) a développé un corpus de comptes-rendus radiologiques pouvant être utilisé pour le développement d'outils de simplification de textes médicaux.

Une autre approche consiste à détecter les composants morphologiques des mots difficiles. Il peut être intéressant de décomposer un terme comme *hématurie* en ses composants *hématie, urine*. Cette décomposition automatique des termes se base souvent sur des méthodes à base de règles ou des approches probabilistes en corpus (Namer, 2003), (Claveau & Kijak, 2014). (Grabar & Hamon, 2015) ont développé un système qui permet d'acquérir des paraphrases non spécialisées pour des termes techniques composés du domaine médical.

3 Notre approche

Dans ce papier, nous étudions la ST d'un type de document médical à savoir les comptes rendus de radiologie. Nous utilisons une méthode de remplacement lexicale non seulement par des synonymes mais aussi par d'autres relations (par exemple, l'hyperonymie). Cette dernière peut être très utile car un terme peut être expliqué comme une incidence spécifique de ses parents. Par exemple, une *carcinome hépatocellulaire* est un *cancer du foie* (relation *is-a*). La base de connaissance sur laquelle se base notre méthode de simplification est le réseau lexico-sémantique JeuxDeMots [4] (JDM) (Lafourcade, 2007).

3.1 Ressources

Deux ressources sont utilisées dans notre approches : (1) une base de connaissances et (2) un corpus de comptes-rendus radiologiques.

Le réseau lexical JeuxDeMots (JDM)

Le réseau JDM est un graphe lexico-sémantique pour la langue française dont les relations entre les termes sont capturées par la combinaison d'un Game With A Purpose (GWAP) (Lafourcade, 2007) avec un outil contributif nommé Diko (contribution manuelle et inférences automatiques avec validation (Zarrouk *et al.*, 2013)). Le réseau contient à la fois des connaissances du domaine général ainsi que des connaissances spécialisées. La partie médicale de la ressource JeuxDeMots a été constituée à partir d'un corpus de comptes rendus cliniques, des pages médicales de wikipédia, du site des maladies rares Orphanet [5] et du Dictionnaire médical de l'Académie de Médecine [6]. En février 2018, le réseau JDM contient 181803113 relations et 2712500 termes. Le tableau suivant (table 1) donne un ordre de grandeur de la quantité d'information que nous avons à notre disposition dans le domaine de la radiologie.

Termes	liens sortants	liens entrants
maladie	16648	21040
anatomie	50 846	94 000
radiologie	733	940

TABLE 1 – Nombre de relations de certains mots clefs dans le réseau JDM.

4. http ://www.jeuxdemots.org/
5. http ://www.orpha.net/consor/cgi-bin/index.php ?lng=FR
6. http ://dictionnaire.academie-medecine.fr/

r_isa termes génériques.
r_synonym synonymes or quasi-synonymes.
r_syn_strict synonymes stricts.
r_equiv acronyme ou abréviation.

FIGURE 1 – Les relations utilisées pour la simplification de textes médicaux

Il existe 80 types de relations dans le réseau mais dans ce travail nous utilisons seulement 4 types différents de relations (figure 1).

Nous utilisons ce réseau car les relations entre les termes sont à la fois pondérées et annotées (Ramadier *et al.*, 2014). Dans le réseau JDM, les relations sont pondérées c'est à dire que le poids reflète la force d'association entre les termes. Par contre, dans le domaine des connaissances spécialisées, la corrélation entre la force d'association de la relation et son importance conceptuelle n'est pas toujours assurée. C'est pourquoi, il est apparu intéressant d'utiliser des annotations pour certaines relations. Ces annotations peuvent nous aider dans la tâche de simplification lexicale (grâce à l'annotation *langage courant*). L'implémentation d'une annotation se fait par réification de la relation à annoter dans le réseau lexical. Le nœud relation créé peut être associé à d'autres termes. L'annotation de relation n'est qu'un type de relation parmi d'autres. Les valeurs d'annotation sont des termes standards (*fréquent, rare, langage courant, etc.*).

Le corpus de comptes-rendus radiologiques

Notre corpus est constitué de 35 000 comptes rendus radiologiques représentant différentes modalités d'imagerie médicale (imagerie par résonance magnétique, scanner, radiographie, échographie, radiologie interventionnelle). Ces comptes rendus sont écrits de façon semi-structurés. Ils sont, en général, divisés en quatre parties, chacune étant écrite de manière non structurée (avec souvent quantité d'acronymes et d'abréviations). Ils ont été, au préalable, dé-identifiés.

3.2 Méthode

Dans un premier temps, il est important d'identifier les mots qui posent le plus de difficulté de compréhension aux patients (Kauchak *et al.*, 2017). Nous sélectionnons les termes difficiles en utilisant une méthode classique, à savoir la fréquence des termes (TF) et la fréquence documentaire (DF) pour calculer l'IDF (Inverse Document Frequency). La reconnaissance des termes composés est effectuée en amont par comparaison au contenu de JDM. Si un terme est polysémique, nous sélectionnons le terme en lien avec la médecine (par exemple, pour kyste nous choisissons le raffinement médecine (*kyste >médecine*) et non celui liés à la botanique (*kyste>botanique*)). Pour choisir un terme du langage courant, nous utilisons les annotations de relations (Ramadier *et al.*, 2014). Si la relation a une annotation *langage courant*, alors le terme médical original est remplacé par son synonyme ou son hyperonyme. Nous présentons un exemple de simplification à la figure 2. Nous remplaçons systématiquement le terme *antérieur* par *en avant* et *postérieur* par *derrière* . Certaines abréviations (*AVP*) seront remplacées par leur signification (*accident de la voie publique*).

Si un terme composé difficile à comprendre n'a pas d'annotation de relations, nous extrayons l'information sémantique des mots qui le composent. En effet, dans le réseau l'information lexicale indique si un mot fait partie du langage courant ou pas.

(a) le patient a une *aphasie* depuis deux jours. → Le patient a un **mutisme** depuis deux jours.
(b) Le patient se plaint de *céphalée* → Le patient se plaint de **maux de tête**
(c) Symptôme : *hématurie* → Symptôme : **sang dans les urines**

FIGURE 2 – Exemples de simplification de terme. Le terme remplacé est en gras.

4 Expérience et résultats

4.1 Expérience

Nous utilisons un sous échantillon de notre corpus (200 comptes rendus) et nous les simplifions grâce à notre approche. Pour l'évaluation manuelle, 250 phrases ont été sélectionnées aléatoirement pour une évaluation humaine manuelle ainsi que pour le test d'évaluation standard pour la compréhension (*cloze test* ou *texte à trou*) (Taylor, 1953). Selon la procédure standard du test, le 5ème mot de chaque phrase est remplacé par un espace blanc. Nous avons recruté 4 personnes (non expertes du domaine médical) pour réaliser le test. Chaque personne réalisait le test sur le compte rendu original et sa version simplifiée. Un expert vérifiait, ensuite, manuellement les réponses pour évaluer leur exactitude. Nous calculions, alors un score (*cloze score* (Zeng-Treitler *et al.*, 2007)) qui représentait le pourcentage de réponses exactes correspondant aux mots effacés.

4.2 Résultats

En moyenne, 10 termes furent simplifiés par compte rendu. La plupart des simplifications (75%) ont été déclarés correctes par l'évaluateur humain. Pour 12% des phrases, le mot remplacé avait un sens légèrement différent par rapport au terme original. Ces erreurs peuvent s'expliquer par le fait que dans certains cas le synonyme n'est pas strict. Par exemple, *kyste* et *abcès* sont synonymes ou quasi-synonymes dans le réseau mais dans le domaine médical, leur sens sont différents. Un autre type d'erreur est que parfois pour la relation d'hyperonymie le mot remplacé était trop général (par exemple, *appendagite* a été remplacé par *maladie*). Nous montrons quelques exemples de termes originaux et leurs synonymes du langage courant (Table 2).

Terme original	**Remplacé par**
aphasie	mutisme
prurit	démangeaison
dyspnée	difficulté à respirer
glioblastome	tumeur maligne du cerveau
arthrite	inflammation des articulations

TABLE 2 – Exemples de termes simplifiés.

La table 3 et table 4 montrent les résultats pour les comptes rendus originaux et simplifiés. Le tableau 3

montre les résultats obtenus avec seulement les annotations de relations (*langage courant*). Le tableau 4 indique les résultats en utilisant à la fois les annotations mais aussi les informations sémantiques présentes dans le réseau JDM.

Comptes rendus originaux	Comptes rendus simplifiés
18%	48%

TABLE 3 – *Cloze score* pour les comptes rendus originaux et simplifiés avec seulement les annotations.

Terme original	Comptes rendus simplifiés
18%	57%

TABLE 4 – *Cloze score* pour les comptes rendus originaux et simplifiés quand on utilise non seulement les annotations mais aussi les informations sémantiques présentes dans le réseau JDM

Le score de 18% sur les comptes rendus originaux prouve que les comptes rendus radiologiques sont difficilement compréhensibles pour les patients.

5 Discussion

Nous décrivons un système de simplification pour un corpus français de comptes rendus radiologiques. Le score (cloze score) pour les textes originaux sont plus faibles que dans d'autres études (Zeng-Treitler *et al.*, 2007) qui traitent des textes médicaux variés (comptes rendus chirurgicaux, lettres de sortie). Les comptes rendus radiologiques sont parmi les plus difficiles à comprendre pour des non-experts. Nous avons implémenté un système basé une simplification lexicale dont le but est d'offrir une meilleure compréhension aux patients. Notre méthode se base sur le réseau JDM et en particulier, sur les annotations de relations pour choisir un terme plus compréhensible. 80% des termes remplacés sont utiles. Si nous utilisons seulement les annotations de relations pour la tâche de simplification, nous obtenons un score de 48%. En utilisant en complément, l'information lexicale présente dans le réseau, nous améliorons notre résultat et nous atteignons un score de 57%. Mais 35% des termes difficiles qui auraient du être remplacés ne l'ont pas été parce qu'il n'y avait pas d'annotations (*langage courant*) dans le réseau. L'évaluation manuelle a aussi montré que le sens original a parfois été légèrement modifié dans certaines phrases. Dans certains cas, les mots ne sont pas strictement synonymes (*œdème* et *gonflement*, par exemple). Nos résultats sont proches de ceux de (Zeng-Treitler *et al.*, 2007) bien que notre corpus soit plus important et ne contienne que des comptes-rendus radiologiques. Notre système est évidement perfectible. Une piste d'amélioration possible est d'augmenter la couverture des annotations au sein du réseau. Nous avons également l'intention de simplifier la syntaxe des phrases dans un deuxième temps.

6 Conclusion

Nous avons développé un système dont l'objectif est d'améliorer la compréhension des comptes-rendus radiologiques par le patient. Les résultats présentés ici restent préliminaires mais sont néanmoins très prometteurs. Dans ce travail, nous avons utilisé le réseau lexico-sémantique JeuxDeMots

comme base de connaissance. Bien que ce réseau soit général, il contient de nombreuses données spécialisées, notamment en médecine et radiologie pouvant être utiles dans le cadre de la tâche de simplification pour ces domaines.

Dans un travail futur, une autre tâche est de simplifier la syntaxe des textes médicaux. Une étude précédente (Campbell & Johnson, 2001) a montré des différences significatives du contenu syntaxique et la complexité entre les rapports de sortie d'hospitalisation et l'anglais quotidien. Une autre étude a souligné la difficulté de simplification syntaxique de textes (Kandula *et al.*, 2010). Pour cette tâche, nous pourrions réaliser une simplification de grammaire (par exemple, les longues phrases pourraient être découpées en phrases plus courtes). Nous planifions aussi de tester notre approche dans d'autres domaines médicaux, comme par exemple l'oncologie, parce que JDM contient des données de ce domaine.

Références

ABRAHAMSSON E., FORNI T., SKEPPSTEDT M. & KVIST M. (2014). Medical text simplification using synonym replacement : Adapting assessment of word difficulty to a compounding language. In *Proceedings of the 3rd Workshop on Predicting and Improving Text Readability for Target Reader Populations (PITR)*, p. 57–65.

CAMPBELL D. A. & JOHNSON S. B. (2001). Comparing syntactic complexity in medical and non-medical corpora. In *Proceedings of the AMIA Symposium*, p.5̃0 : American Medical Informatics Association.

CHAPMAN K., ABRAHAM C., JENKINS V. & FALLOWFIELD L. (2003). Lay understanding of terms used in cancer consultations. *Psycho-Oncology*, **12**(6), 557–566.

CLAVEAU V. & KIJAK E. (2014). Generating and using probabilistic morphological resources for the biomedical domain. In *9th edition of the Language Resources and Evaluation Conference, LREC 2014*, p. 7–p.

G. DIAS, Ed. (2015). *Actes de TALN 2015 (Traitement automatique des langues naturelles)*, Caen. ATALA, HULTECH.

ELHADAD N. & SUTARIA K. (2007). Mining a lexicon of technical terms and lay equivalents. In *Proceedings of the Workshop on BioNLP 2007 : Biological, Translational, and Clinical Language Processing*, p. 49–56 : Association for Computational Linguistics.

GRABAR N. & HAMON T. (2015). Extraction automatique de paraphrases grand public pour les termes médicaux. *Actes de TALN*.

KANDULA S., CURTIS D. & ZENG-TREITLER Q. (2010). A semantic and syntactic text simplification tool for health content. In *AMIA annual symposium proceedings*, volume 2010, p. 366 : American Medical Informatics Association.

KAUCHAK D., LEROY G. & HOGUE A. (2017). Measuring text difficulty using parse-tree frequency. *Journal of the Association for Information Science and Technology*, **68**(9), 2088–2100.

KESELMAN A., LOGAN R., SMITH C. A., LEROY G. & ZENG-TREITLER Q. (2008). Developing informatics tools and strategies for consumer-centered health communication. *Journal of the American Medical Informatics Association*, **15**(4), 473–483.

KESELMAN A., SLAUGHTER L., ARNOTT-SMITH C., KIM H., DIVITA G., BROWNE A., TSAI C. & ZENG-TREITLER Q. (2007). Towards consumer-friendly phrs : patients' experience with

reviewing their health records. In *AMIA Annual Symposium Proceedings*, volume 2007, p. 399 : American Medical Informatics Association.

KESELMAN A. & SMITH C. A. (2012). A classification of errors in lay comprehension of medical documents. *Journal of biomedical informatics*, **45**(6), 1151–1163.

KESKISÄRKKÄ R. (2012). Automatic text simplification via synonym replacement.

KOKKINAKIS D., FORSBERG M., KOKKINAKIS S. J., SMITH F. & ÖHLEN J. (2012). Literacy demands and information to cancer patients. In *International Conference on Text, Speech and Dialogue*, p. 64–71 : Springer.

KVIST M. & VELUPILLAI S. (2013). Professional language in swedish radiology reports–characterization for patient-adapted text simplification. In *Scandinavian Conference on Health Informatics 2013, Copenhagen, Denmark, August 20, 2013*, p. 55–59 : Linköping University Electronic Press.

LAFOURCADE M. (2007). Making people play for lexical acquisition with the jeuxdemots prototype. In *SNLP'07 : 7th international symposium on natural language processing*, p.7̃.

LERNER E. B., JEHLE D. V., JANICKE D. M. & MOSCATI R. M. (2000). Medical communication : do our patients understand ? *The American journal of emergency medicine*, **18**(7), 764–766.

LEROY G., ENDICOTT J. E., MOURADI O., KAUCHAK D. & JUST M. L. (2012). Improving perceived and actual text difficulty for health information consumers using semi-automated methods. In *AMIA Annual Symposium Proceedings*, volume 2012, p. 522 : American Medical Informatics Association.

LEROY G., HELMREICH S. & COWIE J. R. (2010). The influence of text characteristics on perceived and actual difficulty of health information. *International journal of medical informatics*, **79**(6), 438–449.

NAMER F. (2003). Automatiser l'analyse morpho-sémantique non affixale : le système dérif. *Cahiers de grammaire*, **28**, 31–48.

QENAM B., KIM T. Y., CARROLL M. J. & HOGARTH M. (2017). Text simplification using consumer health vocabulary to generate patient-centered radiology reporting : Translation and evaluation. *Journal of medical Internet research*, **19**(12).

RAMADIER L., ZARROUK M., LAFOURCADE M. & MICHEAU A. (2014). Inferring relations and annotations in semantic network : Application to radiology. *Computación y Sistemas*, **18**(3), 455–466.

SLAUGHTER L., KESELMAN A., KUSHNIRUK A. & PATEL V. L. (2005). A framework for capturing the interactions between laypersons' understanding of disease, information gathering behaviors, and actions taken during an epidemic. *Journal of biomedical informatics*, **38**(4), 298–313.

TAYLOR W. L. (1953). "cloze procedure" : A new tool for measuring readability. *Journalism Bulletin*, **30**(4), 415–433.

TRAN T. M., CHEKROUD H., THIERY P. & JULIENNE A. (2009). Internet et soins : un tiers invisible dans la relation médecine/patient. *Ethica Clinica*, **53**, 34–43.

ZARROUK M., LAFOURCADE M. & JOUBERT A. (2013). Inference and reconciliation in a crowdsourced lexical-semantic network. *Computación y Sistemas*, **17**(2).

ZENG-TREITLER Q., GORYACHEV S., KIM H., KESELMAN A. & ROSENDALE D. (2007). Making texts in electronic health records comprehensible to consumers : a prototype translator. In *AMIA Annual Symposium Proceedings*, volume 2007, p. 846 : American Medical Informatics Association.

Algorithmes à base d'échantillonage pour l'entraînement de modèles de langue neuronaux

Matthieu Labeau Alexandre Allauzen
LIMSI, CNRS, Univ. Paris-Sud, Université Paris-Saclay, Campus Universitaire Bât 508, F-91405 Orsay cedex
prénom.nom@limsi.fr

RÉSUMÉ
L'estimation contrastive bruitée (NCE) et l'échantillonage par importance (IS) sont des procédures d'entraînement basées sur l'échantillonage, que l'on utilise habituellement à la place de l'estimation du maximum de vraisemblance (MLE) pour éviter le calcul du softmax lorsque l'on entraîne des modèles de langue neuronaux. Dans cet article, nous cherchons à résumer le fonctionnement de ces algorithmes, et leur utilisation dans la littérature du TAL. Nous les comparons expérimentalement, et présentons des manières de faciliter l'entraînement du NCE.

ABSTRACT
Here the title in English.

Noise Contrastive Estimation (NCE) and Importance Sampling (IS) are sampling based algorithms traditionally used to avoid computing the costly output softmax when training neural language models with Maximum Likelihood Estimation (MLE). In this work, we attempt to summarize how these procedures work, and how they have been used in the computational linguistics literature. We then compare them, and experiment with tricks that ease NCE training.

MOTS-CLÉS : Modèle de langue, Estimation contrastive bruitée, Negative Sampling.

KEYWORDS: Neural Language Model, Noise Contrastive Estimation, Negative Sampling.

Introduction

Les modèles de langue statistiques jouent un rôle essentiel dans la plupart des tâches du TAL, comme la traduction automatique et la reconnaissance de la parole. Les modèles de langue neuronaux (Bengio *et al.*, 2001; Mikolov *et al.*, 2010; Chelba *et al.*, 2014; Józefowicz *et al.*, 2016) ont prouvé leur efficacité, mais il y a un problème commun à la plupart des architectures existantes : la grande taille du vocabulaire de sortie, qui implique de très longs temps de calcul, puisque l'on normalise les scores de sortie en probabilités. Il faut aussi prendre en compte qu'il est très difficile de réaliser des prédictions sur un espace aussi grand.

Une solution de plus en plus utilisée est de réduire la taille du vocabulaire en travaillant non pas avec les mots mais avec des structures contenues dans le mot, comme les morphèmes ou les caractères. Bien que cela puisse être efficace pour certaines applications, on ne fait qu'éviter le problème. D'autres solutions communes sont de réduire ou changer la structure de la couche de sortie, à l'aide de *short-lists* (Schwenk, 2007) ou d'un *softmax hiérarchique* (Morin & Bengio, 2005; Mnih & Hinton, 2009; Le *et al.*, 2011). Les techniques de *self-normalisation* (Devlin *et al.*, 2014; Andreas *et al.*, 2015;

Chen *et al.*, 2016) permettent d'éviter cette normalisation à l'inférence, ce qui l'accélère grandement. Enfin, des méthodes basées sur l'échantillonage comme l'*échantillonage par importance* (Bengio & Sénécal, 2003; Jean *et al.*, 2015) (*Importance Sampling*), et l'*estimation contrastive bruitée* (*Noise contrastive estimation*) (Mnih & Teh, 2012) permettent d'entraîner des modèles de langue neuronaux à grand vocabulaires beaucoup plus rapidement.

Nous cherchons d'abord, dans cet article, à recenser et comparer les plus importantes de ces méthodes et leurs variantes (section 1), et à vérifier pourquoi et comment elles sont utilisées dans la littérature (section 2). Ensuite, nous évaluons ces différents algorithmes, ainsi que des heuristiques de choix des hyperparamètres, sur deux corpus de référence pour l'évaluation des modèles de langue (section 3).

Maximum de vraisemblance et normalisation

L'important temps de calcul nécessaire à l'entraînement des modèles de langue neuronaux vient habituellement de la taille du vocabulaire $\mathcal{V}$. Un modèle défini par ses paramètres θ, produit une distribution P_θ^H multinomiale sur $\mathcal{V}$, conditionné par le contexte d'entrée H. Cette distribution est construite à l'aide de la fonction *softmax* :

$$P_\theta(w|H) = \frac{e^{s_\theta(w,H)}}{\sum_{w' \in \mathcal{V}} e^{s_\theta(w',H)}} = \frac{e^{s_\theta(w,H)}}{Z_\theta(H)} \tag{1}$$

Ici, la fonction $s_\theta(w, H)$, qui dépend de l'architecture du réseau, produit un score pour le couple (H, w). Le dénominateur est la *fonction de partition* $Z_\theta(H)$, qui assure que la distribution de sortie soit normalisée. Ainsi, on peut écrire chaque sortie comme le produit d'une distribution non-normalisée et de l'inverse de la fonction de partition. Habituellement, le modèle est entraîné en minimisant la log-vraisemblance négative de ces distributions conditionelles pour chaque couple $(H, w) \in \mathcal{D}$ présent dans les données.

$$NLL(\theta) = -\sum_{(H,w)\in\mathcal{D}} \log P_\theta(w|H) \tag{2}$$

Pour minimiser cet objectif, on effectue des mises à jour sur les paramètres à l'aide d'une *descente de gradient stochastique* (SGD). Ce gradient est composé de deux termes :

$$\frac{\partial}{\partial\theta} \log P_\theta(w|H) = \frac{\partial}{\partial\theta} s_\theta(w, H) - \sum_{w' \in \mathcal{V}} P_\theta(w'|H) \frac{\partial}{\partial\theta} s_\theta(w', H) \tag{3}$$

Le premier permet l'accroissement de la vraisemblance du mot w, tandis que le second permet de réduire la vraisemblance de tous les autres. Le calcul de la fonction de partition $Z_\theta(H)$ est cependant nécessaire au calcul du second terme, ainsi qu'à celui de la fonction objectif.

1 Échantilloner pour accélerer l'entraînement des modèles de langue neuronaux

Il a été démontré(Gutmann & Hyvärinen, 2013) que la fonction de partition est nécessaire au calcul de la vraisemblance d'un modèle. Paramétriser cette fonction de partition séparément des paramètres θ, comme un paramètre de "mise à l'échelle" pour chaque contexte H, ne peut pas fonctionner, puisque la minimisation de la vraisemblance pourrait se faire indépendamment des données. On s'intéresse donc aux méthodes pour l'entraînement de modèles de langue non normalisés. Les plus populaires sont basées sur l'*échantillonage par importance* (IS) et l'*estimation contrastive bruitée* (NCE).

1.1 Échantillonage par importance

L'idée, explicitée dans (Bengio & Sénécal, 2003), est de ré-écrire le second terme du gradient comme une espérance, que l'on estime à l'aide d'un échantillonage par importance, à l'aide d'une distribution à partir de laquelle on peut échantillonner facilement, P_n, on obtient l'approximation suivante pour le gradient :

$$\frac{\partial}{\partial\theta}\log P_\theta(w|H) \approx \frac{\partial}{\partial\theta}s_\theta(w,H) - \frac{1}{R}\sum_{\substack{i=1\\ \hat{w}_i\sim P_n}}^{k} r_i \frac{\partial}{\partial\theta}s_\theta(\hat{w}_i,H) \text{ avec } r_i = \frac{e^{s_\theta(\hat{w}_i,H)}}{P_n(\hat{w}_i)} \text{ et } R = \frac{1}{k}\sum_{\substack{i=1\\ \hat{w}_i\sim P_n}}^{k} r_i \tag{4}$$

ce qui est un estimateur biaisé pour le gradient de notre objectif. Un problème majeur de cette approche est que les poids r_i peuvent prendre des valeurs très élevées, ce qui augmente grandement la variance de l'estimateur. On peut réduire le biais et la variance en augmentant le nombre d'échantillons k. Cependant, cela allonge le temps de calcul. Pour éviter la croissance de la variance pendant l'entraînement, (Bengio & Sénécal, 2008) cherche à adapter la distribution auxilliaire P_n au fur et à mesure de celui-ci.

Une variante récente, le *target sampling* (Jean *et al.*, 2015), appliquée à la traduction automatique, partitionne les données et définit des sous parties du vocabulaire à partir desquellse échantillonner, qui contiennent au moins les mots cibles de leur partition. L'utilisation de probabilités uniformes sur une sous partie du vocabulaire simplifie grandement l'estimateur, et puisque les mots cibles sont présents dans la distribution, l'estimateur est cohérent. Notre objectif est ainsi équivalent à un softmax que l'on applique seulement aux mots échantillonnés.

1.2 Estimation contrastive bruitée

Le NCE, initialement décrit dans (Gutmann & Hyvärinen, 2010, 2012), permet d'estimer une densité de probabilités paramétrique à partir des données, en considérant la fonction de partition comme un paramètre séparé. Il simule l'estimation par maximum de vraisemblance en apprenant à différencier les exemples provenant des données des exemples échantillonnés d'une *distribution de bruit* auxilliaire P_n. La méthode a été appliquée aux modèles de langue par (Mnih & Teh, 2012) : pour chaque couple

$(H, w) \in \mathcal{D}$, on tire k mots de P_n. Ainsi, les probabilités conditionelles d'un mot w selon H et selon sa classe (C est à 1 si le mot vient des données, et à 0 sinon) sont les suivantes :

$$P(w|C = 1, H) = P_\theta(w|H) \text{ et } P(w|C = 0, H) = P_n(w) \tag{5}$$

ce qui nous donne les probabilités postérieures d'appartenance aux classes :

$$P(C = 1|w, H) = \frac{P_\theta(w|H)}{P_\theta(w|H) + kP_n(w)} \text{ et } P(C = 0|w, H) = \frac{kP_n(w)}{P_\theta(w|H) + kP_n(w)} \tag{6}$$

Chacune des distribution $P_\theta(w|H)$ peut-être ré-écrite comme le produit d'une distribution non normalisée $p_\theta(w|H) = e^{s_\theta(w,H)}$ et d'un paramètre Z_H dépendant de H, représentant la fonction de partition. Dans (Mnih & Teh, 2012), les auteurs remarquent qu'il est difficile d'apprendre un paramètre par contexte H possible, et que si l'on se contente de fixer ces paramètres à 1, les distributions non normalisées se normalisent d'elles-même [1]. On appelle ce processus l'*auto-normalisation*. L'objectif de classification est obtenu en maximisant la log-vraisemblance de l'appartenance de l'exemple des données w à la classe $C = 1$ et des exemples $(\hat{w}_i)_{1 \leq i \leq k}$ tirés de P_n à $C = 0$. On obtient ainsi, pour un exemple :

$$J_\theta^H(w) = \log \frac{p_\theta(w|H)}{p_\theta(w|H) + kP_n(w)} + \sum_{i=1}^{k} \log \frac{kP_n(\hat{w}_i)}{p_\theta(\hat{w}_i|H) + kP_n(\hat{w}_i)} \tag{7}$$

L'un des avantages du NCE sur l'IS est que les poids utilisés sont compris entre 0 et 1. Concernant le choix de la distribution de bruit, il est démontré(Gutmann & Hyvärinen, 2010, 2012) que l'erreur d'estimation des paramètres θ est asymptotiquement indépendante de P_n lorsque le nombre d'exemples de bruit échantillonnés par exemple venant des données k est assez grand. Ainsi, le choix de P_n revient à un compromis entre une distribution plus proche de celle des données, ou une distribution plus simple à partir de laquelle on peut facilement échantillonner, et ainsi utiliser un k grand. (Mnih & Teh, 2012) compare l'utilisation des distributions uniforme et unigramme, obtenant des résultats bien plus précis avec la seconde.

Le *BlackOut* ((Ji *et al.*, 2015)) est une approche plus récente, similaire à la fois aux NCE et IS. On peut le considérer comme un NCE avec une distribution de bruit particulière, obtenue en ajustant le poids des échantillons à l'aide de ratios similaire à ceux utilisés en IS. Comme pour le NCE, on peut utiliser des distributions non normalisées. Le principal avantage du BlackOut est que les poids utilisés introduisent une dépendance au contexte H dans la probabilité de bruit, ce qui nous donne une distribution plus proche des données à un faible coût computationnel.

1.2.1 Simplification : *Negative sampling*

La méthode du *negative sampling* (NS) a été popularisée par l'algorithme du skip-gram utilisé pour l'entraînement de word embeddings(Mikolov *et al.*, 2013), et bien que très proche du NCE, elle ne permet pas d'optimiser directement la vraisemblance d'un modèle de langue(Dyer, 2014). Cependant, (Melamud *et al.*, 2016, 2017) ont montré qu'il est tout de même viable d'entraîner un modèle de

1. Ce qui s'explique par le grand nombre de paramètres libres dans les modèles neuronaux

langue avec le NS. Comme pour le NCE, on échantillonne k exemples $(\hat{w}_i)_{1\leq i\leq k}$ d'une distribution de bruit. L'objectif est simple : il maximise la vraisemblance de la régression logistique qui différencie données du bruit. Ainsi, la fonction de score de notre modèle $s_\theta(w|H)$ ne paramétrise pas la log-densité de probabilité des données mais le log-ratio de cette densité avec celle de la distribution de bruit.

$$J_\theta^H(w) = \log \sigma(s_\theta(w|H)) + \sum_{i=1}^{k} \log \sigma(-s_\theta(\hat{w}_i|H)) \tag{8}$$

Et on obtient l'estimation de la probabilité conditionelle par la simple opération :

$$\hat{P}(w|H) \propto P_n(w) \exp s_\theta(w|H) \tag{9}$$

2 Utilisations dans la littérature

Le NCE principalement été utilisé dans le contexte de la traduction automatique : (Vaswani *et al.*, 2013; Baltescu & Blunsom, 2015) utilisent une distribution de bruit unigramme, tandis que (Zoph *et al.*, 2016) utilise une distribution de bruit uniforme. Le NCE a aussi été appliqué à la reconnaissance de la parole (Chen *et al.*, 2015), avec la distribution unigramme. Dans ces travaux, les auteurs ont fixé un paramètres représentant la log-fonction de partition à 9 (au lieu de 1) pour faciliter l'entraînement.

Malgré les garanties théoriques offertes par le NCE, (Chen *et al.*, 2016) ont démontré qu'il peut être inconsistant lorsqu'il est utilisé avec de très grands vocabulaires, avec des perplexités très différentes pour des modèles ayant convergé vers des valeurs de la fonction objectif similaires. D'autres travaux (Józefowicz *et al.*, 2016) mettent en évidence que le NCE est moins efficace que l'IS [2].

3 Comparaisons et heuristiques de paramétrage

Nous proposons de comparer les méthodes présentées ici, d'abord avec un modèle que l'on supervise attentivement, sur un corpus de taille réduite, puis avec un modèle classique sur un corpus plus grand, pour lequel maximiser la vraisemblance ne serait pas une option, car trop coûteux.

Nous faisons l'hypothèse que les difficultées liées à l'utilisation du NCE proviennent de l'auto-normalisation, qui peut être vue comme une tâche supplémentaire à effectuer en parallèle par le modèle. Nous expérimentons donc avec des heuristiques qui visent à faciliter ce processus. Ainsi, on commencera par ajouter aux méthodes présentées un modèle entraîné avec le NCE, mais dont nous normalisons le score $p_\theta(w|H) = e^{s_\theta(w,H)}$ en $P_\theta(w|H)$. Cela n'a que peu d'intérêt en pratique, mais cela nous permet d'évaluer la difficulté du modèle a s'entraîner avec une distribution non normalisée. Le résultat obtenu par le NCE normalisé peut ainsi servir de référence à laquelle comparer les heuristiques proposées.

2. Notons tout de même que l'implémentation de l'IS utilisée est légèrement différente de celle décrite dans (Bengio & Sénécal, 2003) et a des points communs avec celle du target sampling.

3.1 Heuristiques proposées

La distribution de bruit la plus couramment utilisée avec les algorithmes présentés est la distribution unigramme, que nous utiliserons ici aussi. Cependant, dans certain cas (Mikolov *et al.*, 2013), cette distribution est mise à la puissance $\alpha = 0.75$, ce qui permet de la lisser. Intuitivement, cela permet d'échantilloner plus souvent les mots plus rares, et donc de faciliter leur apprentissage. Toutefois, la distribution de bruit s'éloigne de celle des données, ce qui est censé rendre l'algorithme moins efficace.

Ensuite, suivant une idée présentée dans (Chen *et al.*, 2015), nous pouvons, au lieu de fixer le paramètre censé représenter la fonction de partition Z_c à 1, utiliser une valeur fixée plus proche de sa valeur à l'initialisation (qui dépend de la taille du vocabulaire). En pratique, cela revient à entraîner le modèle avec les distributions $p_\theta(w|H) = e^{s_\theta(w,H)}/Z_c$. Nous adoptons deux approches : d'abord, nous fixons $Z_c = |\mathcal{V}|$. Ainsi, nous aidons le modèle à s'auto-normaliser, en le plaçant à l'initialisation dans un état ou il est presque normalisé. Puis, nous apprenons la valeur de Z_c comme un paramètre du modèle. Dans ce cas, le modèle apprendra en parallèle un paramètre indépendant du contexte qui l'aidera à se normaliser tout au long de l'entraînement.

3.2 Approche expérimentale

Le corpus de taille réduite sur lequel nous comparons nos modèles est le Penn Tree Bank (PTB), muni de la totalité de son vocabulaire d'entraînement, c'est à dire environ $44K$ mots. Nous entraînons ces modèles avec $k = 100$ échantillons tirés de la distribution de bruit, lorsque cela s'applique. Nos modèles de langue sont des modèles récurrents (LSTM), ici avec 2 couches cachées de dimension 300. Les fonctions d'activations sont des *Rectified Linear Units* (Relu). Nous utilisons un algorithme de SGD classique, avec un learning rate de 1.0, et nous limitons la valeur des gradients à 5.0. Nous entraînons ces modèles pendant un minimum de 30 époques et un maximum de 60, et ne sauvegardons les époques suivantes que lorsque le modèle améliore sa performance sur l'ensemble de validation. Nous arrêtons l'entraînement lorsque que la perplexité de test n'a pas été améliorée pendant 10 époques consécutives.

Le second corpus utilisé est le WMT 1 Billion Word Benchmark (Chelba *et al.*, 2014). Les modèles sont un peu plus conséquents (la dimension de la couche cachée est 512), et nous utilisons l'optimiseur Adam (Kingma & Ba, 2014). Puisque le volume de données est beaucoup plus important, l'entraînement se fait sur entre 1 et 2 époques, Nous arrêtons ici l'entraînement lorsque que la perplexité de test n'a pas été améliorée après un nombre d'exemples équivalent au tiers d'une époque.

3.3 Résultats

Les meilleures perplexités obtenues sur les données de test des deux corpus sont présentées dans la table 1. On peut constater que les résultats obtenus avec la méthode du maximum de vraisemblance (MLE) et le NCE normalisé sont presque équivalents. Cependant, les résultats du NCE non normalisé sont de loin inférieurs, ce qui montre la difficulté pour le modèle de se normaliser lui même. Le NS est plus efficace que le NCE, mais est moins précis que l'IS, qui obtient le meilleur résultat. Lisser la distribution de bruit est efficace pour le NCE et le NS, mais ne réduit que de peu l'écart avec l'IS. Enfin, fixer ou apprendre Z_c donne de bons résultats avec le NCE. Notamment, le fait d'apprendre Z_c

Méthode	Penn TreeBank (PTB)	WMT 1B-word benchmark
MLE	150.2	-
NCE Normalisé	159.3	-
NCE	306.0	X
IS	168.3	77.9
NS	228.3	102.4
NCE + $\alpha = 0.75$	277.0	X
IS + $\alpha = 0.75$	171.5	67.2
NS + $\alpha = 0.75$	195.8	83.7
NCE + $Z_c = \|\mathcal{V}\|$	178.6	72.8
NCE + $Z_c \in \theta$	172.3	70.9

TABLE 1 – Meilleures perplexités de test obtenues respectivement sur le corpus PTB, avec un vocabulaire d'entraînement complet, et sur le WMT 1 Billion Word Benchmark, avec un vocabulaire de 64K mots. 'X' indique que l'algorithme n'a pas donné une perplexité inférieure à $|\mathcal{V}|$.

permet au NCE d'approcher le résultat obtenu avec l'IS.

Ces tendances se vérifient sur le second corpus : alors que les méthodes impliquant une normalisation sont inutilisables pour un corpus de cette taille, le NCE ne converge pas sur la durée de l'entraînement choisi. Mais le NS converge, et l'IS est ici aussi bien plus précis. Lisser la distribution améliore la perplexitée obtenue par l'IS, qui donne le meilleur résultat. Enfin, apprendre automatiquement Z_c permet ici aussi au NCE d'approcher la performance de l'IS.

4 Conclusion

Étant donné la contrainte imposée par la taille du vocabulaire sur le temps d'entraînement des modèles de langue neuronaux, nous nous sommes attachés à recenser et comparer des méthodes d'entraînement basées sur l'échantillonnage. Ces méthodes évitent l'étape coûteuse de normalisation liée à l'objectif du maximum de vraisemblance. Nous avons montré que, comme décrit dans la littérature, l'*Échantillonage par importance* est plus efficace que l'*estimation contrastive bruitée*, qui est difficile à utiliser. Cependant, nous avons montré qu'avec un paramétrage judicieux de la fonction de partition, cette dernière méthode peut se révéler très efficace.

Références

ANDREAS J., RABINOVICH M., JORDAN M. I. & KLEIN D. (2015). On the accuracy of self-normalized log-linear models. In *Advances in Neural Information Processing Systems 28 : Annual Conference on Neural Information Processing Systems 2015, December 7-12, 2015, Montreal, Quebec, Canada*, p. 1783–1791.

BALTESCU P. & BLUNSOM P. (2015). Pragmatic neural language modelling in machine translation. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics : Human Language Technologies*, p. 820–829, Denver, Colorado : Association for Computational Linguistics.

BENGIO Y., DUCHARME R. & VINCENT P. (2001). A neural probabilistic language model.

BENGIO Y. & SÉNÉCAL J.-S. (2003). Quick training of probabilistic neural nets by importance sampling. In *Proceedings of the conference on Artificial Intelligence and Statistics (AISTATS)*.

BENGIO Y. & SÉNÉCAL J.-S. (2008). Adaptive importance sampling to accelerate training of a neural probabilistic language model. *IEEE Trans. Neural Networks*, **19**(4), 713–722.

CHELBA C., MIKOLOV T., SCHUSTER M., GE Q., BRANTS T., KOEHN P. & ROBINSON T. (2014). One billion word benchmark for measuring progress in statistical language modeling. In *INTERSPEECH 2014, 15th Annual Conference of the International Speech Communication Association, Singapore, September 14-18, 2014*, p. 2635–2639.

CHEN W., GRANGIER D. & AULI M. (2016). Strategies for training large vocabulary neural language models. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1 : Long Papers)*, p. 1975–1985, Berlin, Germany : Association for Computational Linguistics.

CHEN X., LIU X., GALES M. J. F. & WOODLAND P. C. (2015). Recurrent neural network language model training with noise contrastive estimation for speech recognition. In *ICASSP*, p. 5411–5415 : IEEE.

DEVLIN J., ZBIB R., HUANG Z., LAMAR T., SCHWARTZ R. & MAKHOUL J. (2014). Fast and robust neural network joint models for statistical machine translation. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1 : Long Papers)*, p. 1370–1380, Baltimore, Maryland : Association for Computational Linguistics.

DYER C. (2014). Notes on noise contrastive estimation and negative sampling. *CoRR*, **abs/1410.8251**.

GUTMANN M. & HYVÄRINEN A. (2010). Noise-contrastive estimation : A new estimation principle for unnormalized statistical models. In *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics, AISTATS 2010, Chia Laguna Resort, Sardinia, Italy, May 13-15, 2010*, p. 297–304.

GUTMANN M. & HYVÄRINEN A. (2012). Noise-contrastive estimation of unnormalized statistical models, with applications to natural image statistics. *Journal of Machine Learning Research*, **13**, 307–361.

GUTMANN M. & HYVÄRINEN A. (2013). Estimation of unnormalized statistical models without numerical integration. In *Proceedings of the Workshop on Information Theoretic Methods in Science and Engineering*.

JEAN S., CHO K., MEMISEVIC R. & BENGIO Y. (2015). On using very large target vocabulary for neural machine translation. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1 : Long Papers)*, p. 1–10, Beijing, China : Association for Computational Linguistics.

JI S., VISHWANATHAN S. V. N., SATISH N., ANDERSON M. J. & DUBEY P. (2015). Blackout : Speeding up recurrent neural network language models with very large vocabularies. *CoRR*, **abs/1511.06909**.

JÓZEFOWICZ R., VINYALS O., SCHUSTER M., SHAZEER N. & WU Y. (2016). Exploring the limits of language modeling. *CoRR*, **abs/1602.02410**.

KINGMA D. P. & BA J. (2014). Adam : A method for stochastic optimization. *CoRR*, **abs/1412.6980**.

LE H.-S., OPARIN I., ALLAUZEN A., GAUVAIN J.-L. & YVON F. (2011). Structured output layer neural network language model. p. 5524–5527, Prague, Czech Republic.

MELAMUD O., DAGAN I. & GOLDBERGER J. (2016). PMI matrix approximations with applications to neural language modeling. *CoRR*, **abs/1609.01235**.

MELAMUD O., DAGAN I. & GOLDBERGER J. (2017). A simple language model based on pmi matrix approximations. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, p. 1861–1866, Copenhagen, Denmark : Association for Computational Linguistics.

MIKOLOV T., KARAFIÁT M., BURGET L., CERNOCKÝ J. & KHUDANPUR S. (2010). Recurrent neural network based language model. In *INTERSPEECH 2010, 11th Annual Conference of the International Speech Communication Association, Makuhari, Chiba, Japan, September 26-30, 2010*, p. 1045–1048.

MIKOLOV T., SUTSKEVER I., CHEN K., CORRADO G. S. & DEAN J. (2013). Distributed representations of words and phrases and their compositionality. In C. J. C. BURGES, L. BOTTOU, M. WELLING, Z. GHAHRAMANI & K. Q. WEINBERGER, Eds., *Advances in Neural Information Processing Systems 26*, p. 3111–3119. Curran Associates, Inc.

MNIH A. & HINTON G. E. (2009). A scalable hierarchical distributed language model. In D. KOLLER, D. SCHUURMANS, Y. BENGIO & L. BOTTOU, Eds., *Advances in Neural Information Processing Systems 21*, p. 1081–1088. Curran Associates, Inc.

MNIH A. & TEH Y. W. (2012). A fast and simple algorithm for training neural probabilistic language models. In *Proceedings of the 29th International Conference on Machine Learning, ICML 2012, Edinburgh, Scotland, UK, June 26 - July 1, 2012*.

MORIN F. & BENGIO Y. (2005). Hierarchical probabilistic neural network language model. In R. G. COWELL & Z. GHAHRAMANI, Eds., *Proceedings of the Tenth International Workshop on Artificial Intelligence and Statistics*, p. 246–252 : Society for Artificial Intelligence and Statistics.

SCHWENK H. (2007). Continuous space language models. *Comput. Speech Lang.*, **21**(3), 492–518.

VASWANI A., ZHAO Y., FOSSUM V. & CHIANG D. (2013). Decoding with large-scale neural language models improves translation. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, p. 1387–1392, Seattle, Washington, USA : Association for Computational Linguistics.

ZOPH B., VASWANI A., MAY J. & KNIGHT K. (2016). Simple, fast noise-contrastive estimation for large rnn vocabularies. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics : Human Language Technologies*, p. 1217–1222, San Diego, California : Association for Computational Linguistics.

Étude Expérimentale d'Extraction d'Information dans des Retranscriptions de Réunions

Pegah ALIZADEH [1] Peggy CELLIER [2] Thierry CHARNOIS [3]
Bruno CRÉMILLEUX [1] Albrecht ZIMMERMANN[1]
(1) Normandie Univ., UNICAEN, ENSICAEN, CNRS – UMR GREYC, Caen, France
(2) Univ Rennes, CNRS, INSA, IRISA - UMR 6074, F-35000 RENNES, FRANCE
(3)LIPN-UMR CNRS 7030, PRES SORBONNE PARIS-CITÉ, FRANCE
peggy.cellier@irisa.fr, thierry.charnois@lipn.univ-paris13.fr,
{pegah.alizadeh, bruno.cremilleux, albrecht.zimmermann}@unicaen.fr

RÉSUMÉ
Nous nous intéressons dans cet article à l'extraction de thèmes à partir de retranscriptions textuelles de réunions. Ce type de corpus est bruité, il manque de formatage, il est peu structuré avec plusieurs locuteurs qui interviennent et l'information y est souvent éparpillée. Nous présentons une étude expérimentale utilisant des méthodes fondées sur la mesure *tf-idf* et l'extraction de *topics* sur un corpus réel de référence (le corpus AMI) pour l'étude de réunions. Nous comparons nos résultats avec les résumés fournis par le corpus.

ABSTRACT
An Experimental Approach For Information Extraction in Multi-Party Dialogue Discourse

In this paper, we address the task of information extraction for meeting transcripts. The meeting documents are not usually well-structured and lacks of formatting and punctuation while the information are distributed over multiple sentences. We investigate on the use of numerical statistic or topic modeling methods on a real dataset containing multi-part dialogue texts. We evaluate our experiments with respect to the summaries provided in the dataset.

MOTS-CLÉS : Extraction d'information, corpus de dialogue, détection de thèmes.

KEYWORDS: Information Extraction, Dialogue Texts, Topic Modeling.

1 Introduction

La plupart des gens passent une grande quantité de leur temps en réunion. Une fois celle-ci terminée, il reste encore un compte rendu à produire, rendant compte des principaux aspects abordés lors de la réunion, comme les problèmes rencontrés et les décisions prises. Il est aujourd'hui possible d'enregistrer et de stocker une réunion en audio voire en vidéo. À partir de ces enregistrements, plusieurs outils dits *"Speech-to-text"* [1] permettent ensuite de générer une retranscription textuelle de tout ce qui a été dit lors de la réunion. Un enjeu important est alors d'être capable d'extraire automatiquement de ces retranscriptions textuelles, souvent fortement bruitées, des informations et des résumés facilitant la création du compte rendu de la réunion. Le projet REUs (relevant de l'appel

1. Exemple d'outil : http ://www.vocapia.com

à projets FUI 22), dans lequel s'inscrit ce travail, a pour but ultime de générer automatiquement un compte-rendu de réunion entre plusieurs humains à partir de la retranscription textuelle de celle-ci.

Quelques travaux se sont intéressés à l'extraction d'information dans les retranscriptions de réunions. Le système le plus abouti est celui présenté par (Tur *et al.*, 2010, 2008) qui fournit un système d'analyse de réunions appelé CALO. CALO retranscrit automatiquement les minutes de la réunion en texte puis différentes parties de la réunion sont identifiées et annotées : les thèmes, les tours de parole, les actions, les décisions et deux résumés (abstractif et extractif) sont fournis. Pour détecter les thèmes et segmenter le texte, CALO s'appuie sur un modèle génératif de thèmes proche de la méthode LDA (Latent Drichlet Allocation) (Purver *et al.*, 2006). À l'aide d'une approche non supervisée, une segmentation de la réunion est produite et permet de connaître simultanément de "quoi" les participants parlent (i.e., les thèmes) à la réunion et "quand" (i.e., segments). Malgré une approche non-supervisée, la méthode nécessite de faire des hypothèses sur la distribution des thèmes et le nombre de segments dans le texte de la réunion. Ce genre d'hypothèses limite l'utilisation de l'approche sur des applications réelles. En ce qui concerne l'extraction d'actions et de décisions, CALO utilise une approche structurelle. Les différentes prises de paroles de la réunion sont classées en fonction de leur rôle dans le processus : "définition de tâche", "accord" et "acceptation d'une responsabilité". Puis les actions (Purver *et al.*, 2007) et les décisions (Fernández *et al.*, 2008) sont détectées. Enfin en ce qui concerne la tâche de génération d'un résumé, CALO extrait différentes versions réduites du texte et choisit celle qui a la meilleure valeur de la mesure ROUGE (Riedhammer *et al.*, 2008) par rapport à des résumés donnés par un oracle.

Dans la littérature, d'autres approches se sont intéressées à ces problématiques ou de façon plus générale à l'analyse de discours. Il existe des approches de segmentation de texte comme (Galley *et al.*, 2003; Georgescul *et al.*, 2007) qui s'appuient sur les changements de distribution lexicale. D'autres approches comme (R. Fernández & Peters, 2008) extraient des mots importants par rapport à un problème en utilisant des classifieurs et des modèles de séquences et en s'appuyant sur les informations lexicales. Les approches de segmentation ou d'extraction de mots sont des méthodes intéressantes mais nécessitent ensuite de déterminer le type de chaque segment (thèmes, décisions, tours de parole, etc). De façon plus générale, plusieurs travaux s'intéressent à la détection d'événements notamment dans les réseaux sociaux (Sayyadi *et al.*, 2009; He *et al.*, 2007). Toutefois les retranscriptions de réunions n'ont pas les mêmes caractéristiques que les textes issus des réseaux sociaux, en particulier dans le rapport au temps. Beaucoup de tweets sont produits par unité de temps dans les réseaux sociaux, alors qu'une seule intervention est produite par tour de parole dans une réunion.

Dans le contexte du projet REUs, afin d'alimenter le compte-rendu devant être produit, plusieurs informations doivent être extraites comme des événements, des décisions, des problèmes, des solutions, un résumé. Les difficultés pour construire un tel système sont multiples. Par exemple, comment détecter qu'une décision importante par rapport à la réunion est prise ? Comment associer une information temporelle à un sujet d'une réunion ou mettre en relation différentes facettes d'un même événement ? Ces tâches ne sont pas simples et elles sont encore plus aiguës dans le contexte de retranscriptions de réunions car les données sont bruitées, les propos ne sont pas toujours structurés, plusieurs locuteurs interviennent rendant plus complexe le fil des propos. Dans cet article, nous nous intéressons à une des étapes nécessaire à la réalisation d'un système générant un compte-rendu de réunions : l'extraction, à partir des retranscriptions textuelles, des thèmes (en anglais *topics*) qui ont été discutés pendant la réunion. Plus particulièrement, nous présentons une étude expérimentale de méthodes classiques d'extraction d'information et de topics que nous avons menée sur le corpus AMI (Carletta, 2007; AMI, 2010), un corpus de référence pour l'étude des réunions. Nous discutons des différentes particularités de ce type de corpus (interactions multiples très variables, flux de

mots non structurés, ponctuation faible, formatage, etc) et de leurs conséquences sur le processus d'extraction d'information.

2 Extraction d'information dans une retranscription de réunion

L'objectif est d'extraire des informations utiles pour un compte-rendu de réunion. Nous nous sommes donc intéressés aux approches permettant d'extraire des mots "importants" dans un texte afin de voir leur comportement sur des retranscriptions de réunions. Nous avons considéré deux approches bien connues : la mesure *tf-idf* et une approche de *topic modeling*, qui sont rappelées ici.

Fréquence du terme - Fréquence inverse du document (*tf-idf*) La Fréquence du terme - Fréquence inverse du document (*tf-idf*) est une technique qui donne les poids les plus élevés aux termes (i.e. mots) qui apparaissent les plus fréquemment dans un document par rapport aux autres documents du corpus. La fréquence d'un terme (tf(w, d)) est simplement le nombre d'occurrences du mot w dans le document d. La fréquence inverse du document (idf) est la proportion de documents du corpus dans laquelle le mot w apparaît. Plus précisément : $idf(w, D) = \log \frac{N}{1+|d \in D \ s.t. \ w \in d|}$ où N est le nombre total de documents du corpus. Le dénominateur est le nombre de documents dans lesquels un mot w apparaît dans le corpus D. La mesure *tf-idf* est ainsi définie par : tfidf$(w, d, D) =$ tf(w, d)idf(w, d)

***Topic Modeling* : Factorisation par matrices non négatives** Il existe plusieurs approches de *topic modeling* permettant de définir des thèmes apparaissant dans un corpus. Il y a des approches probabilistes telles que l'approche LDA (Latent Dirichlet Allocation) (Blei *et al.*, 2003) mais aussi des méthodes s'appuyant sur l'agèbre linéaire comme *NMF* (Factorisation par matrices non négatives) (Lee & Seung, 1999). Dans cet article nous avons choisi d'utiliser la méthode *NMF* car d'autres expériences non décrites ici ont montré que *NMF* donnait dans ce cas de meilleurs résultats. Dans l'approche *NMF*, on représente m mots apparaissant dans n documents via une matrice $\mathbf{A} \in \mathbb{R}^{m \times n}$. L'objectif de la méthode *NMF* est ensuite de décomposer $\mathbf{A}$ en deux matrices $\mathbf{W}$ et $\mathbf{H}$ telles que $\mathbf{A} \sim \mathbf{WH}$. Les colonnes de la matrice $\mathbf{W} \in \mathbb{R}^{m \times k}$ sont les thèmes (*topics*) identifiés et les lignes les mots représentant ces thèmes. Chaque case de la matrice représente le poids pour un mot d'être relié à un thème. La matrice $\mathbf{H} \in \mathbb{R}^{k \times n}$ représente la façon dont un document est en relation avec k thèmes.

3 Corpus AMI

Le corpus de réunions AMI (Carletta, 2007) contient 100 heures de réunion enregistrées en utilisant plusieurs appareils d'enregistrement synchronisés. Tous les participants aux réunions parlent anglais. Pour certains l'anglais est leur langue maternelle et pour d'autres non. Cela représente 171 réunions qui se regroupent en deux types : les **réunions à base de scénario** (*scenario-based meetings*) et les **réunions sans scénario** (*non-scenario based meetings*). Les **réunions à base de scénario** font partie de séries de 3 à 4 réunions simulant les réunions de quatre participants devant concevoir un système. Les réunions d'une même série ont toutes lieu le même jour. Les **réunions sans scénario** sont de vraies réunions qui ont été enregistrées. Le corpus AMI a été transcrit et annoté avec des informations concernant les entités nommées présentes et les différents tours de parole. Pour chaque réunion un résumé abstractif et un résumé extractif sont fournis. Le résumé abstractif d'une réunion contient environ 200 mots et consiste en un texte libre donnant un résumé général de la réunion ainsi que des

explications à propos des décisions prises et des problèmes évoqués lors de la réunion. Le résumé extractif identifie des morceaux du texte de la réunion qui couvrent les informations contenues dans le résumé abstractif. Notons qu'un élément du résumé abstractif peut faire référence à plusieurs tours de parole et qu'un tour de parole peut être relié à plusieurs éléments du résumé abstractif.

Disponibilité du corpus Une version segmentée du corpus AMI (Carletta, 2007; AMI, 2010) est disponible en ligne [2]. Toutefois le corpus mis à disposition est au format NXT (Nite XML Toolkit), qui ne permet pas de faire des traitements sur le texte. Nous avons recréé les fichiers textes de retranscription et les avons mis à disposition en ligne [3].

4 Extraction d'information dans le corpus AMI

Procédure d'expérimentation Comme mentionné précédemment, le corpus AMI contient deux types de meetings : les réunions avec scénario et sans scénario. Nous nous sommes concentrés sur les réunions avec scénario car elles représentent 138 réunions sur 171. De plus, la plupart des réunions sans scénario n'ont pas de résumé disponible, ce qui ne permet pas d'évaluation.

Pour chacune des deux expériences nous appliquons une approche : *tf-idf* ou *NMF*. Le résultat de l'approche est ensuite évalué en comparant les informations extraites des réunions aux informations présentes dans les deux types de résumé (abstractif et extractif). La précision par rapport à un résumé abstractif (respectivement extractif) est donnée par la formule suivante : $\frac{\|\{w \in R\} \cap \{w \in S\}\|}{\|\{w \in R\}\|}$ où R est l'ensemble des mots du résultat et S les mots du résumé considéré (abstractif ou extractif). Notons que l'on applique une racinisation sur les mots et c'est la racine qui est prise en compte pour l'appariement de mots. Par exemple, "painting" et "paint" sont considérés comme le même mot.

4.1 Comparaison entre séries de réunions

Dans cette première expérience, nous regroupons toutes les réunions d'une même série en un seul corpus. Il y a donc 34 corpus, chacun représentant une série de réunions. Ensuite nous extrayons les informations propres à chacun des corpus selon les deux méthodes.

tf-idf Avec l'approche utilisant la mesure *tf-idf*, nous calculons les 20 mots les plus importants pour chaque série de réunions. La table 1 (première colonne) donne des exemples de mots extraits avec cette mesure pour 2 séries de réunions avec scénario [4]. On voit que pour la série 1 le terme avec le plus haut score par rapport aux autres mots de la réunion est "matthew", suivi de "mael".

À la figure 1, le schéma du haut donne le score de précision par rapport aux résumés par abstraction et par extraction. Comme attendu, la précision pour le résumé extractif (plus de 60%) est meilleure que pour le résume abstractif (moins de 40%). En effet, cette mesure calcule le pourcentage de mots présents dans chaque résumé. Le résumé extractif est plus long et est un sous-ensemble du texte initial contrairement au résumé abstractif qui est une reformulation de ce qui a été dit dans la réunion.

2. http://groups.inf.ed.ac.uk/ami/download/
3. https://github.com/pegahani/AMI-prep
4. Tous les résultats pour toutes les séries sont disponibles ici : https://github.com/pegahani/Event_detection/blob/master/result/result_4_block_scen.txt

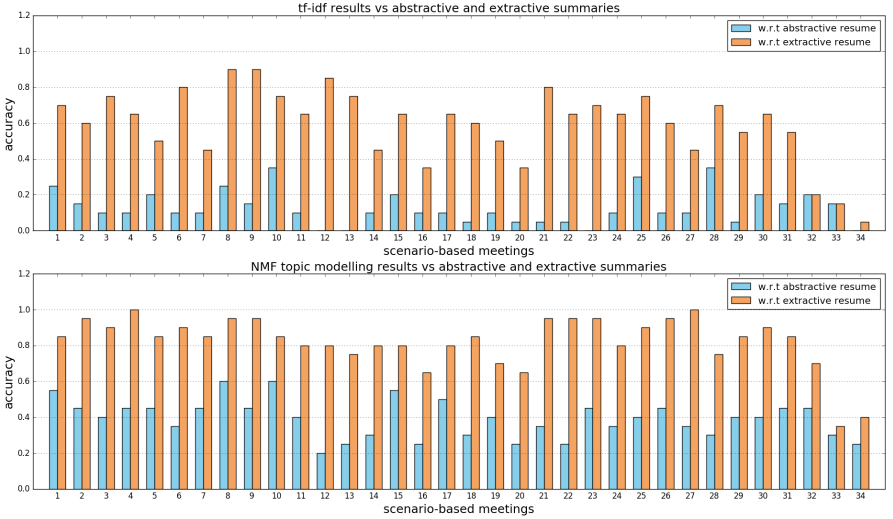


FIGURE 1 – Le graphique du haut montre la précision de l'approche *tf-idf* par rapport aux résumés abstractif et extractif. Le graphique du bas montre la précision de l'approche *NMF* par rapport aux résumés abstractif et extractif.

Série de réunions	Mots extraits avec *tf-idf*	Mots extraits avec *NMF*
Série 1	**'matthew', 'mael', 'anna'**, 'exce', 'ip', 'doctor', 'decline', 'assemble', 'streamed', 'customizing', 'asian', 'voter', 'undes', 'nanne', 'highperformance', 'fik', 'protec', 'underlie', 'provin', 'zebras'	keys, **matthew**, browse, innovation, functionalities, v_c_r_, **mael, anna**, sixteen, perfect, demographic, r_c_, receiver, surf, present blinking, cents, store, movie, presented
Série 4	**'mushroom', 'jordan'**, 'coarse', 'baba', 'alimentation', 'mush', 'gestures', 'kemy', 'institute', 'frahan', 'florent', 'laser', 'longmund', 'sleeping', 'ada', 'saucer', 'trois', 'ecological', 'hmmm', 'eatable'	controller, **mushroom**, gesture, google, pineapple, powerful, david, base, wireless, traditional, lemon, **jordan**, wooden, sophisticated, wire, vocal, participant, ball, bulb, recognise

TABLE 1 – Informations extraites pour 2 séries de réunions avec scénario.

Topic Modeling Nous avons aussi mené une expérience avec une approche *topic Modeling*. Pour cela, nous avons utilisé la méthode *NMF* (cf section 2) pour affecter un thème à chaque réunion avec scénario. Comme nous l'avons signalé précédemment dans cette section, il y a 34 réunions de ce type, nous utilisons donc le modèle *NMF* en fixant à 34 le nombre de thèmes à classifier. Pour implémenter l'approche *NMF*, nous utilisons le package gensim (Řehůřek & Sojka, 2010). Après l'application de *NMF* sur les réunions, un thème est affecté à chaque réunion. Ce thème est représenté par les 20 mots les plus "importants". Sur le tableau 1, les mots extraits pour 2 réunions sont indiqués dans la colonne de gauche.

Comme pour la méthode s'appuyant sur le *tf-idf*, dans le graphe du bas de la figure 1 la précision de l'approche *NMF* sur les 34 corpus est donnée. Nous constatons que l'approche *NMF* donne de meilleurs résultats que la comparaison soit faite avec le résumé abstractif ou extractif. Nous pensons que les meilleurs résultats de *NMF* viennent du fait que dans cette approche tous les mots de toutes les réunions sont considérés pour l'extraction des thèmes. Avec l'approche s'appuyant sur *tf-idf* seuls les mots de la série de réunions qui sont peu fréquents dans les autres réunions vont être extraits. On peut ainsi manquer des mots/thèmes importants.

M_1	M_2	M_3	M_4
25 : animal	9 : age, lunch, teletext	8 : solar, wood	17 : criteria
19 : cat	8 : percent, young	6 : titanium	13 : seven
12 : favourite	6 : pay	5 : spongy, concepts	12 : evaluation
11 : tool, dog	5 : settings, zap, seventy, users	4 : dark, sample, doublecurved, materials, sensor, banana, circuit, cases, vegetables, fruit	9 : sample
7 : training, draw	4 : set, messages, group, mode, infrared, recognition, speech		8 : special, false
6 : rabbit, profit, fish			6 : evaluate, process
4 : friendly, bird, tail, whiteboard, width, characteristics, elephant, morning			5 : prototype, leadership, creativity, under
			4 : scale, single, fifteen, team, average, budget, curve

TABLE 2 – Mots les plus souvent répétés et ayant les plus hauts scores *tf-idf*.

4.2 Caractérisation de réunions individuelles

Comme indiqué précédemment, les réunions d'AMI sont destinées à simuler un projet, par exemple la conception d'une télécommande, du début à la fin. Toutes les réunions liées à une série (c'est-à-dire des réunions jouées par le même groupe de participants) sont condensées en une seule journée. Il y a 34 séries différentes. Les différentes séries de réunions portent sur le même scénario mais varient dans la mise en œuvre concrète du script. Si les réunions d'une même série comportent des caractéristiques communes, nous nous attendons à ce qu'elles apparaissent dans les termes extraits par *tf-idf*.

Pour vérifier notre hypothèse : nous traitons une série de réunions comme un corpus et identifions pour chaque réunion individuelle dans cette série les 20 mots avec les scores plus élevés selon *tf-idf*. Pour chacun de ces mots, nous comptons alors le nombre de fois où il est apparu dans l'ensemble des mots dérivés de la même étape dans les autres séries. Par exemple, nous rassemblons les ensembles de mots dérivés de la première réunion de chaque série (M_1) et comptons les doublons. La table 2 montre les mots répétés plus que 4 fois pour chaque étape (10% arrondi)[5]. On voit que le terme «animal» a été utilisé dans 73.5% de toutes les premières réunions.

La table 2 (y compris 10% des vingt mots les plus élevés pour l'ensemble des 34 séries) montre que la plupart des mots sont apparus dans la première ou la dernière réunion de la série alors que les deux réunions intermédiaires ont une petite part. Par exemple, dans l'étape M_1, «animal» est répété 29 fois, mais dans l'étape M_2, moins de mots apparaissent, par exemple «age» est apparu 9 fois. Nous voyons que la première et la dernière réunions d'une série ont des mots plus liés que les deuxième et troisième réunions. Cela montre que la première rencontre semble toujours être un brainstorming lié aux animaux et la quatrième réunion sur la façon d'évaluer le succès du projet. Lors de la troisième réunion, la discussion a porté sur les matériaux à utiliser. Enfin, la présence de «lunch» en tant que mot le plus répété lors de la deuxième réunion montre que cette réunion a eu lieu en fin de matinée.

4.3 Comparaison entre réunions d'une même série

Dans une seconde expérience, nous extrayons les informations propres à une réunion par rapport aux autres réunions de la même série. Nous avons testé les deux méthodes : *tf-idf* et *NMF*. Contrairement à l'expérience précédente, la méthode s'appuyant sur *tf-idf* a donné dans nos expériences de meilleurs résultats que celle s'appuyant sur *NMF*. Nous supposons que cela ait du au fait que les corpus de cette expérience sont plus courts.

5. Pour les résultats complets, voir : `https://github.com/pegahani/Event_detection/blob/master/result/result_4_4.txt`

Aux figures 2 et 3, les scores de précision obtenus pour chacune des réunions de chacune des séries par rapport au résumé abstractif ou extractif sont donnés. Si l'on compare les résultats par rapport aux deux types de résumés, on retrouve la même tendance que sur l'expérience précédente portant sur les séries de réunion (section 4.1) : les scores des résumés par extraction sont meilleurs (la plupart au dessus de 0.6) que pour les résumés par abstraction (au dessous de 0.6).

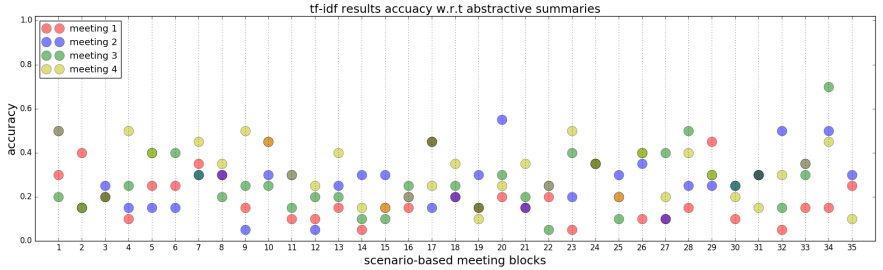


FIGURE 2 – Précision, par rapport au résumé abstractif, des thèmes extraits par *tf-idf*.

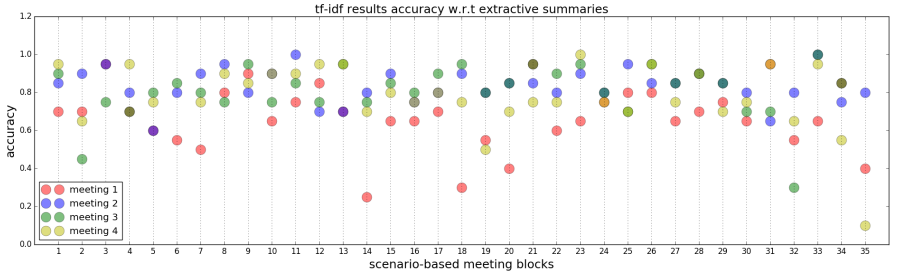


FIGURE 3 – Précision, par rapport au résumé extractif, des thèmes extraits par *tf-idf*.

5 Conclusion

Ce travail représente une étude de base. L'objectif est d'alimenter un outil de génération automatique de compte-rendu de réunion. Nous avons mené des expériences sur l'extraction d'information dans des corpus de transcription de réunions en testant deux approches de l'état-de-l'art sur un corpus de référence AMI. Les résultats obtenus nous semblent encourageants sur ce type de corpus et justifient l'utilisation d'approches robustes. Ils montrent aussi que les deux approches sont complémentaires puisque l'approche *topic modeling* (*NMF*) donne de meilleurs résultats pour la caractérisation de réunions individuelles, alors que pour la comparaison de réunions d'une même série c'est l'approche *tf-idf*. Nous souhaitons par la suite travailler sur des méthodes permettant d'affiner l'information extraite.

Remerciements Ce travail est soutenu par le FUI 22 (projet REUs).

Références

AMI (2010). Augmented multi-party interaction. `http://www.amiproject.org`. [Online].

BLEI D. M., NG A. Y. & JORDAN M. I. (2003). Latent dirichlet allocation. *J. Mach. Learn. Res.*, **3**, 993–1022.

CARLETTA J. (2007). Unleashing the killer corpus : experiences in creating the multi-everything AMI meeting corpus. *Language Resources and Evaluation*, **41**, 181–190.

FERNÁNDEZ R., FRAMPTON M., EHLEN P., PURVER M. & PETERS S. (2008). Modelling and detecting decisions in multi-party dialogue. In *Proceedings of the 9th SIGdial Workshop on Discourse and Dialogue*, SIGdial '08, p. 156–163, Stroudsburg, PA, USA : Association for Computational Linguistics.

GALLEY M., MCKEOWN K., FOSLER-LUSSIER E. & JING H. (2003). Discourse segmentation of multi-party conversation. In *Proceedings of the 41st Annual Meeting on Association for Computational Linguistics - Volume 1*, ACL '03, Stroudsburg, PA, USA : Association for Computational Linguistics.

GEORGESCUL M., CLARK A. & ARMSTRONG S. (2007). Exploiting structural meeting-specific features for topic segmentation. In *TALN/RECITAL*, p. 15–24, Toulouse (France).

HE Q., CHANG K. & LIM E.-P. (2007). Analyzing feature trajectories for event detection. In *Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '07, p. 207–214, New York, NY, USA : ACM.

LEE D. D. & SEUNG H. S. (1999). Learning the parts of objects by nonnegative matrix factorization. *Nature*, **401**, 788–791.

PURVER M., DOWDING J., NIEKRASZ J., EHLEN P., NOORBALOOCHI S. & PETERS S. (2007). Detecting and summarizing action items in multi-party dialogue. In *In Proc. of the 9th SIGdial Workshop on Discourse and Dialogue*.

PURVER M., GRIFFITHS T. L., KÖRDING K. P. & TENENBAUM J. B. (2006). Unsupervised topic modelling for multi-party spoken discourse. In *Proceedings of the 21st International Conference on Computational Linguistics and the 44th Annual Meeting of the Association for Computational Linguistics*, ACL-44, p. 17–24, Stroudsburg, PA, USA : Association for Computational Linguistics.

R. FERNÁNDEZ, M. FRAMPTON J. D. A. A. P. E. & PETERS S. (2008). Identifying relevant phrases to summarize decisions in spoken meetings. In *Proceedings of Interspeech'08*, Brisbane.

ŘEHŮŘEK R. & SOJKA P. (2010). Software Framework for Topic Modelling with Large Corpora. In *Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks*, p. 45–50, Valletta, Malta : ELRA. `http://is.muni.cz/publication/884893/en`.

RIEDHAMMER K., FAVRE B. & HAKKANI-TÜR D. (2008). Packing the Meeting Summarization Knapsack. In *Interspeech, Brisbane (Australia)*, Unknown, Unknown or Invalid Region.

SAYYADI H., HURST M. & MAYKOV A. (2009). Event detection and tracking in social streams. In *In Proceedings of the International Conference on Weblogs and Social Media (ICWSM 2009). AAAI*.

TUR G., STOLCKE A., VOSS L., DOWDING J., FAVRE B., FERNANDEZ R., FRAMPTON M., FRANDSEN M., FREDERICKSON C., GRACIARENA M., HAKKANI-TUR D., KINTZING D., LEVEQUE K., MASON S., NIEKRASZ J., PETERS S., PURVER M., RIEDHAMMER K., SHRIBERG E., TIEN J., VERGYRI D. & YANG F. (2008). The calo meeting speech recognition and understanding system. In *2008 IEEE Spoken Language Technology Workshop*, p. 69–72.

TUR G., STOLCKE A., VOSS L., PETERS S., HAKKANI-TUR D., DOWDING J., FAVRE B., FERNANDEZ R., FRAMPTON M., FRANDSEN M., FREDERICKSON C., GRACIARENA M., KINTZING D., LEVEQUE K., MASON S., NIEKRASZ J., PURVER M., RIEDHAMMER K., SHRIBERG E., TIEN J., VERGYRI D. & YANG F. (2010). The calo meeting assistant system. *IEEE Transactions on Audio, Speech, and Language Processing*, **18**, 1601–1611.

Analyse morpho-syntaxique en présence d'alternance codique

José Carlos Rosales Núñez Guillaume Wisniewski
LIMSI, CNRS, Univ. Paris-Sud, Université Paris-Saclay, 91 405 Orsay, France
prénom.nom@limsi.fr

RÉSUMÉ
L'alternance codique est le phénomène qui consiste à alterner les langues au cours d'une même conversation ou d'une même phrase. Avec l'augmentation du volume généré par les utilisateurs, ce phénomène essentiellement oral, se retrouve de plus en plus dans les textes écrits, nécessitant d'adapter les tâches et modèles de traitement automatique de la langue à ce nouveau type d'énoncés. Ce travail présente la collecte et l'annotation en partie du discours d'un corpus d'énoncés comportant des alternances codiques et évalue leur impact sur la tâche d'analyse morpho-syntaxique.

ABSTRACT
PoS tagging of Code Switching

Code switching (CS) is a phenomenon consisting in alternating languages during a conversation or within a sentence. Due to the increasing volume of User Generated Content, code switching, that used to be mainly an oral phenomenon, is becoming more and more present in written texts, creating the need to adapt NLP tasks and models to this new type of content. This work presents the collection and annotation of a corpus containing CS sentences and assesses the impact of code switching on PoS tagging.

MOTS-CLÉS : Erreur d'annotation, analyse morpho-syntaxique, adaptation au domaine.

KEYWORDS: Annotation error, PoS-tagging, domain adaptation.

1 Introduction

Le *code-switching* (CS) ou alternance codique est le phénomène qui consiste à alterner les langues au cours d'une même conversation (Isurin *et al.*, 2009; Myers-Scotton, 1997). C'est un phénomène fréquent chez les locuteurs des communautés bilingues et multilingues qui ont l'habitude de passer d'une langue à l'autre au cours d'une conversation et parfois même à l'intérieur d'une phrase (Auer, 1998). La table 1 donne plusieurs exemples d'énoncés produits par des locuteurs anglais-espagnol illustrant ce phénomène.

Le code-switching est un phénomène oral que l'on ne retrouve quasiment pas à l'écrit : la quasi totalité des corpus existants (comme, par exemple, (Özlem Çetinoğlu, 2016; Ramanarayanan & Suendermann-Oeft, 2017)) est constituée de transcriptions. Mais avec l'augmentation du volume de contenu généré par les utilisateurs *(user generated content)* notamment sur les différents média sociaux (Facebook, Twitter, ...) ou les forums, de plus en plus de textes écrits comportent des énoncés écrits en plusieurs langues. En effet, par de nombreux aspects, les contenus générés par les utilisateurs ont des caractéristiques qui se rapprochent de ceux de la langue parlée. La collecte d'énoncés présentant des alternances codiques se retrouve donc simplifiée (il n'est plus nécessaire d'enregistrer et de

Conversation	◇ mi entonces ahorayou want to speak Spanish ! ◇ and we 're like " are ... I 'm sure this is like como unos chinitos ahí trabajan " ◇ no mentiradothat was a day one five dollars .
Twitter	◇ I used to think his name was Toño 😐 until they told me it was Jonny ☺ I was like pos Como Se Llame /.- 😜 ♡ ◇ The fact that Jonny already knew me I yo no 😉 It 's like Baby Porke Nunca me hablabas 😀 😀

TABLE 1: Exemples d'énoncés prononcés par des locuteurs anglais-espagnol comportant une alternance codique. Les mots anglais sont en bleu, les mots espagnols en rouge, les ponctuations, entités nommées et autres symboles en noir. Les données sont issues des deux corpus décrits à la section 2.

transcrire des dialogues), ouvrant la possibilité de nouvelles études. Mais, ce développement nécessite également l'adaptation des méthodes et des tâches existantes à ce nouveau type de données.

Ce travail comporte deux contributions : nous décrivons, dans un premier temps (§2), la collecte d'un nouveau corpus d'énoncés comportant des alternances codiques et leur annotation en partie du discours. Nous évaluerons ensuite l'impact de ce phénomène sur l'analyse morpho-sytaxique (§3).

2 Collecte et annotation des corpus

Nous allons considérer, dans nos expériences, deux corpus d'énoncés produits par des locuteurs bilingues espagnol-anglais correspondant aux deux types d'énoncés CS mentionnés dans l'introduction : la langue parlée et les contenus générés par l'utilisateur.

Le premier corpus que nous utilisons repose sur le corpus `Miami Bangor`[1], l'un des plus gros corpus de transcription contenant des alternances codiques : il est constitué des transcriptions de plus de 35h d'entretiens avec 84 locuteurs bilingues de la région de Miami. Les phrases de ce corpus ont été segmentées en mots automatiquement. Des annotateurs humains ont ensuite annoté chaque mot du corpus pour indiquer quelle était sa langue et son étiquette morpho-syntaxique, en suivant le guide d'annotation du projet UD (Nivre *et al.*, 2017). Une description complète de cette campagne d'annotation est faite dans (Soto & Hirschberg, 2017). Dans nos expériences, seules les phrases contenant un changement de langues ont été conservées. Dans la suite de cet article nous appellerons ce corpus `Conversation`.

Le second corpus est issu de la campagne d'évaluation organisée dans le cadre du second atelier *Computational Approaches to Linguistic Code Switching* (Molina *et al.*, 2016). Cette campagne avait pour objectif d'identifier la langue dont chaque mot d'un twee était issu. Comme le `Miami Bangor`, ce corpus comporte des énoncés mélangeant anglais et espagnol. Pour collecter ceux-ci, les organisateurs de la campagne ont ciblé les comptes Twitter d'utilisateurs habitant des régions dans lesquelles sont présents de nombreux locuteurs bilingues (en pratique, New-York et Miami) et qui suivent le compte Twitter de radios espagnoles. Les tweets collectés ont été segmentés et étiquetés semi-manuellement pour indiquer à quelle langue chaque mot appartenait.

À partir de cette information de langue, nous avons étiqueté automatiquement les corpus en utilisant

1. `http://bangortalk.org.uk/speakers.php?c=miami`

	n. phrases	n. mots	longueur phrase	% mots anglais	% mots espagnol	% symboles
Conversation	2 980	36 677	12 mots	39,0%	46,1%	14,9%
Twitter	1 002	15 474	14 mots	50,6%	28,8%	20,5%

TABLE 2: Principales caractéristiques des corpus utilisés dans ce travail. Les symboles correspondent à tous les mots dont il est impossible d'identifier la langue (noms propres, ponctuation, émoticône, ...).

des dictionnaires extraits de Wiktionary et des corpus anglais et espagnol du projet UD. Le guide d'annotation du projet UD a été étendu pour ajouter deux étiquettes correspondant aux hastags et aux émoticône. Deux annotateurs [2] ont ensuite vérifié et corrigé manuellement l'ensemble des étiquettes. Ce corpus sera appelé Twitter dans le reste de cet article. [3]

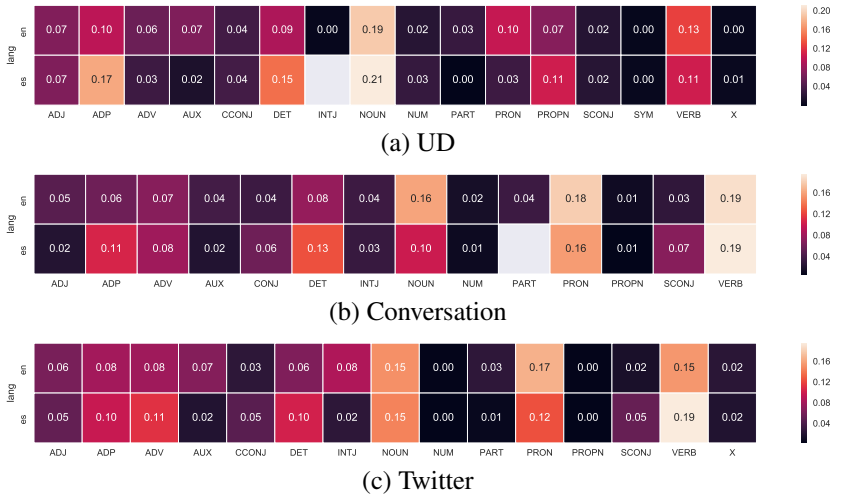
La table 2 résume les principales caractéristiques de ces deux corpus. Ces statistiques montrent que le corpus Twitter présente une alternance codique plus faible : la majorité des phrases ne comporte que quelques mots en espagnol et une large majorité de mots en anglais. En pratique, sur les deux corpus, environ 45% des phrases, il n'y a qu'un seul mot qui n'est pas exprimé dans la langue majoritaire, ce qui suggère que les deux corpus comportent de nombreux cas d'*emprunt lexical* et n'est pas constitué uniquement d'alternance codique à proprement parler (Myers-Scotton, 1997).

Pour caractériser les phénomènes d'alternance codique, nous avons considéré la distribution des étiquettes morpho-syntaxiques par langue à l'intérieur de chaque corpus analysé, résultat présenté dans la Figure 1. En comparant les distributions, il apparait clairement que, comme on pouvait s'y attendre, les corpus présentent des données de nature très différente (p. ex. la proportion d'adjectif varie considérablement d'un corpus à un autre). La langue majoritaire, c'est-à-dire, celle qui est la plus utilisée dans le corpus (espagnol pour Conversation et anglais pour Twitter) ne semble, par contre, ne pas avoir d'impact sur la nature des mots prononcés dans une langue ou dans l'autre.

3 Analyse morpho-syntaxique en présence d'alternance codique

Analyseur morpho-syntaxique à base d'historique Nous utilisons un analyseur morpho-syntaxique à base d'historique (Black *et al.*, 1992; Tsuruoka *et al.*, 2011). Dans ces approches, la prédiction d'une séquence d'étiquettes morpho-syntaxiques est modélisée sous la forme d'une suite de problèmes de décision, consistant chacun à prédire l'étiquette d'une observation. Chaque décision est prise par un classifieur multi-classe utilisant comme descripteurs des informations extraites de la structure d'entrée, ainsi que les décisions prises antérieurement. Nous utilisons, dans toutes nos expériences, un perceptron moyenné comme classifieur multi-classe (Collins, 2002). Nous utilisons des caractéristiques simples que l'on retrouve, à notre connaissance, dans tous les étiqueteurs morpho-syntaxique : mots courants, mots dans une fenêtre de ± 2, étiquettes des deux mots précédents (et leur conjonction), conjonction du mot courant et de l'étiquette précédente, ... [4] Une description détaillée

2. Un des annotateurs est un locuteur natif de l'espagnol ; les deux annotateurs parlent couramment l'anglais.
3. Ce corpus est téléchargeable librement à partir des pages personnelles des deux auteurs.
4. Les entrées sont également transformées : tous les nombres, les URL, les émoticônes et les mentions sont remplacées par un même token



(a) UD

(b) Conversation

(c) Twitter

FIGURE 1: Distribution des étiquettes morpho-syntaxiques selon les langues sur les corpus `UD` (haut), `Conversation` (milieu) et `Twitter` (bas).

de ce modèle se trouve dans (Wisniewski *et al.*, 2014b,a).

Les performances de ce modèle sont légèrement inférieures aux performances d'un modèle d'analyse morpho-syntaxique neuronal tout en étant beaucoup plus rapide à entrainer (notamment à cause du nombre réduit d'hyper-paramètres) : par exemple, sur les corpus anglais et espagnol du projet *Universal Dependencies*, notre modèle obtient, respectivement, une précision de 93,5% et 95,0% alors que le modèle UDPIPE (Straka *et al.*, 2016) obtient 93,5% et 95,5%.

Adaptation du modèle pour l'alternance codique Nous proposons dans cette section une modification très simple du modèle que nous venons de présenter visant à prendre en compte l'alternance de langues dans une phrase. Le principal objectif de ce modèle est de permettre de caractériser et de quantifier les problèmes soulevés par la présence d'alternance codique dans des phrases.

La méthode proposée repose sur une spécialisation du classifieur utilisé dans notre analyseur morpho-syntaxique. Elle consiste simplement à identifier la langue de chaque mot et à utiliser deux classifieurs différents, chacun adapté à une des langues en présence, pour prendre les décisions successives lors de l'inférence.

Plus précisément, nous apprenons, indépendamment, deux analyseurs morpho-syntaxiques : le premier sur un corpus anglais étiqueté avec des informations morpho-syntaxique, le second sur un corpus similaires en espagnol. Ces corpus sont identiques à ceux utilisés pour l'apprentissage d'un analyseur « classique ». Lors de l'inférence, en fonction de la langue du mot dont on cherche à prédire l'étiquette, l'un ou l'autre des modèles est utilisé pour réaliser la prédiction. Bien que les étiquettes soient prédites par des modèles indépendants (au sens où aucun information n'est partagée entre les langues au moment de l'apprentissage), l'historique est partagé. Par souci de simplification, l'identification de la langue d'un mot est réalisée de manière indépendante.

Protocole expérimental Le modèle introduit dans le paragraphe précédent a été testé sur les deux corpus introduits dans la section 2. Les modèles monolingues sont appris sur les corpus `UD_English` et `UD_Spanish` du projet UD. [5]

Identification de la langue Les langues de chaque mot du corpus de test sont prédites à l'aide de l'outil `langid.py` avec ses modèles pré-entrainés (Lui & Baldwin, 2012). Cet outil repose sur un classifieur bayésien naïf et est capable d'identifier la langue d'un *document* de manière très précise la langue d'un document, mais sa capacité à prédire la langue d'un mot unique n'a, à notre connaissance, jamais été évaluée. Nous utilisons également avec un modèle à réseaux de neurones (2 couches cachées composé de 128 et 64 neurones et une couche de sortie 'sotfmax') construit spécifiquement pour prédire la langue d'un mot sans connaissance du contexte dans lequel il a été utilisé. Ce réseau considère en entrée une représentation « one-hot » des 4-grams de lettres du mot ou le mot entier si sa longeur est plus petite que 4. Ce modèle est entrainé sur les deux corpus UD considérés.

Lorsque les mots sont pris isolément (c.-à-d. sans considérer leur context), L'outil `langid.py` est capable de prédire correctement la langue d'un mot issus d'un corpus présentant des alternances codiques dans 56,8% des cas, tandis que notre modèle atteint un taux de reconnaissance de 93,3% de reconnaissance. Cette grosse différence de performances est très certainement lié au fait que `langid.py` a besoin de plusieurs mots de la même langue regroupés dans une phrase pour identifier avec certitude la langue d'une phrase et ne peut donc être utilisé à un niveau sous-phrastique. Le modèle à base de *n*-gram de lettres que nous avons développé ne souffre pas de cette limite.

Résultats expérimentaux Les performances de notre modèle d'analyse morpho-syntaxique sont comparées à trois modèles de référence : deux analyseurs syntaxiques appris uniquement sur les modèles monolingues (c.-à-d. un analyseur appris uniquement sur le corpus anglais et un autre appris uniquement sur le corpus espagnol) ainsi qu'un analyseur appris lorsque les phrases du corpus anglais sont mélangées aux phrases du corpus espagnol. Nous considérons également, comme point de comparaison, un résultat oracle correspondant à une situation dans laquelle la langue est systématiquement correctement identifiée.

La Table 3 rapporte le taux d'erreur obtenu par chacun de ces modèles sur les deux corpus considérés. Ces taux d'erreurs sont moyennés sur 10 apprentissages.

Les résultats très faibles obtenus par les modèles monolingues montrent qu'il est clairement nécessaire de prendre en compte la présence d'alternance codique. Une simple concaténation des corpus monolingues semblent par contre déjà permettre une réduction forte du nombre d'erreurs, la différence avec les performances obtenues sur les corpus de test de l'UD pouvant s'expliquer par la nature des données considérées : l'UD contient essentiellement des textes journalistiques et issus de wikipédia alors que les corpus `Conversation` et `Twitter` contiennent de la parole spontanée.

De manière très surprenante, la méthode par spécialisation du modèle que nous proposons obtient des résultats légèrement plus mauvais que la simple concaténation des corpus, même lorsque la langue est connue de manière certaine. Ce résultat montre que la connaissance de la langue d'un mot n'apporte pas une information pertinente à la prédiction de son étiquette morpho-syntaxique.

5. Au final, les corpus présentant de l'alternance codique ne sont utilisés que pour l'évaluation de notre modèle : nous apprenons deux modèles d'analyse morpho-syntaxique indépendant, l'un sur le corpus UD espagnol et l'autre sur le corpus UD anglais. Lors de la phase de test, ces deux modèles sont utilisés alternativement en fonction du résultat de l'identification de la langue de chaque mot.

méthode	Conversation	Twitter	UD espagnol	UD anglais
Analyseur anglais	45,5%	23,8%	67,5%	6,5%
Analyseur espagnol	39,0%	60,2%	5,0%	69,7%
Analyseur anglais+espagnol	13,2%	18.2%	5,1%	7,1%
Sélection (`langid`)	37,4%	30,5%	—	—
Sélection	17,3%	25,8%	—	—
Sélection oracle	13,1%	19,7%	—	—

TABLE 3: Taux d'erreurs obtenus par les différents modèles sur les deux corpus considérés.

Plusieurs raisons peuvent expliquer ce résultat. En particulier, le nombre de mots identiques en anglais et en espagnol et dont l'étiquette morpho-syntaxique diffère n'est peut-être pas suffisant pour avoir un impact sur le taux d'erreur global. En pratique, sur les ensembles d'apprentissage des corpus `UD_English` et `UD_Spanish` il n'y en a que 2 424 types communs (pour 16 568 mots anglais et 44 739 mots espagnols) et seulement 583 d'entre eux ont des catégories morpho-syntaxique différente dans les deux langues. De plus, l'annotation des langues semble ne pas toujours être de très bonne qualité et présente de nombreuses décisions arbitraires (par exemple, au niveau des interjections et des noms propres). Il faut également noter que le modèle anglais+espagnol est appris sur un corpus deux fois plus grand que les modèles monolingues.

Comme on pouvait s'y attendre, les performances chutent de manière significative lorsque la langue d'un mot est déterminée de manière automatique : lorsque la langue est prédite par `langid.py`, les taux d'erreur obtenus sur les corpus `Twitter` et `Conversation` sont, respectivement, de 30,5% et de 37,4%. En utilisant notre classifieur à réseau de neurones pour la détection de la langue, le taux d'erreur est de 17,3% pour le corpus `Conversation` et 25,8% sur le corpus `Twitter`.

4 Conclusion

Nous avons présenté dans ce travail deux corpus contenant des énoncés avec de l'alternance codique et annotés en partie du discours. C'est, à notre connaissance, l'une des première fois qu'un aussi gros volume de données présentant ce phénomène est annoté avec des informations morpho-syntaxiques ce qui ouvre la voie à beaucoup de perspectives pour analyser ce phénomène.

Nous avons également présenté des modèles d'analyse morpho-syntaxiques simples, mais conçus pour prendre en compte les phénomènes d'alternance codique et analyser leurs performances. Ces résultats montrent la difficulté de la tâche.

Remerciements

Ces travaux ont été en partie financés par l'Agence Nationale de la Recherche (projet PARSITI, ANR-16-CE33-0021). Nous remercions les relecteurs pour leurs commentaires et suggestions.

Références

P. AUER, Ed. (1998). *Code-Switching in Conversation : Language, Interaction and Identity*. Routledge.

BLACK E., JELINEK F., LAFFERTY J., MAGERMAN D. M., MERCER R. & ROUKOS S. (1992). Towards history-based grammars : Using richer models for probabilistic parsing. In *Proceedings of the Workshop on Speech and Natural Language*, HLT'91, p. 134–139, Stroudsburg, PA, USA : Association for Computational Linguistics.

COLLINS M. (2002). Discriminative training methods for hidden markov models : Theory and experiments with perceptron algorithms. In *Proceedings of the 2002 Conference on Empirical Methods in Natural Language Processing*, p. 1–8 : Association for Computational Linguistics.

ISURIN L., WINFORD D. & DE BOT K. (2009). *Multidisciplinary Approaches to Code Switching*. John Benjamins Publishing.

LUI M. & BALDWIN T. (2012). langid.py : An off-the-shelf language identification tool. In *Proceedings of the ACL 2012 System Demonstrations*, p. 25–30, Jeju Island, Korea : Association for Computational Linguistics.

MOLINA G., ALGHAMDI F., GHONEIM M., HAWWARI A., REY-VILLAMIZAR N., DIAB M. & SOLORIO T. (2016). Overview for the second shared task on language identification in code-switched data. In *Proceedings of the Second Workshop on Computational Approaches to Code Switching*, p. 40–49, Austin, Texas : Association for Computational Linguistics.

MYERS-SCOTTON C. (1997). *Duelling Languages : Grammatical Structure in Codeswitching*. Clarendon Press.

NIVRE J., AGIĆ Ž., AHRENBERG L. & OTHER (2017). Universal dependencies 2.1. LINDAT/CLARIN digital library at the Institute of Formal and Applied Linguistics (ÚFAL), Faculty of Mathematics and Physics, Charles University.

RAMANARAYANAN V. & SUENDERMANN-OEFT D. (2017). Jee haan, i'd like both, por favor : Elicitation of a code-switched corpus of hindi-english and spanish-english human-machine dialog. In *Interspeech 2017, 18th Annual Conference of the International Speech Communication Association, Stockholm, Sweden, August 20-24, 2017*, p. 47–51.

SOTO V. & HIRSCHBERG J. (2017). Crowdsourcing universal part-of-speech tags for code-switching. In *Interspeech 2017, 18th Annual Conference of the International Speech Communication Association, Stockholm, Sweden, August 20-24, 2017*, p. 77–81.

STRAKA M., HAJIČ J. & STRAKOVÁ J. (2016). UDPipe : trainable pipeline for processing CoNLL-U files performing tokenization, morphological analysis, POS tagging and parsing. In *Proceedings of the 10th International Conference on Language Resources and Evaluation (LREC 2016)*, Portorož, Slovenia : European Language Resources Association.

TSURUOKA Y., MIYAO Y. & KAZAMA J. (2011). Learning with lookahead : Can history-based models rival globally optimized models ? In *Proceedings of the Fifteenth Conference on Computational Natural Language Learning*, CoNLL'11, p. 238–246, Portland, Oregon, USA : Association for Computational Linguistics.

WISNIEWSKI G., PÉCHEUX N., GAHBICHE-BRAHAM S. & YVON F. (2014a). Cross-lingual part-of-speech tagging through ambiguous learning. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, p. 1779–1785, Doha, Qatar : Association for Computational Linguistics.

WISNIEWSKI G., PÉCHEUX N., KNYAZEVA E., ALLAUZEN A. & YVON F. (2014b). Apprentissage partiellement supervisé d'un étiqueteur morpho-syntaxique par transfert cross-lingue. In *Proceedings of TALN 2014 (Volume 1 : Long Papers)*, p. 173–183, Marseille, France : Association pour le Traitement Automatique des Langues.

ÖZLEM ÇETINOĞLU (2016). A turkish-german code-switching corpus. In N. C. C. CHAIR), K. CHOUKRI, T. DECLERCK, S. GOGGI, M. GROBELNIK, B. MAEGAARD, J. MARIANI, H. MAZO, A. MORENO, J. ODIJK & S. PIPERIDIS, Eds., *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, Paris, France : European Language Resources Association (ELRA).

Simplification de schémas d'annotation : un aller sans retour ?

Cyril Grouin
CNRS, LIMSI
Université Paris-Saclay
F-91405 Orsay
cyril.grouin@limsi.fr

RÉSUMÉ

Dans cet article, nous comparons l'impact de la simplification d'un schéma d'annotation sur un système de repérage d'entités nommées (REN). Une simplification consiste à rassembler les types d'entités nommées (EN) sous deux types génériques (personne et lieu), l'autre revient à mieux définir chaque type d'EN. Nous observons une amélioration des résultats sur les deux versions simplifiées. Nous étudions également la possibilité de retrouver le niveau de détail des types d'EN du schéma d'origine à partir des versions simplifiées. L'utilisation de règles de conversion permet de recouvrer les types d'EN d'origine, mais il reste une forme d'ambiguïté contextuelle qu'il est impossible de lever au moyen de règles.

ABSTRACT

Annotation scheme simplification : a one way trip with no return ?

In this paper, we study the impact of annotation scheme simplification on named entity recognition (NER) performances. One simplification consists in merging all named entity (NE) types into two main types (person and location), while the other simplification relies on a better definition of all NE types. We achieved better results on the two simplified versions of the annotation scheme. We also study the ability to recover the original NE types from the simplified versions. The use of post-processing rules allows to recover a few original NE types. Nevertheless, we faced with a kind of contextual ambiguity which seems hard to process using rules.

MOTS-CLÉS : Entités nommées, schéma d'annotation, simplification.

KEYWORDS: Named entities, annotation scheme, simplification.

1 Introduction

L'annotation de corpus est un processus long et coûteux, mais utile pour construire des ressources, réaliser des analyses linguistiques, entraîner des modèles d'apprentissage statistique, ou pour évaluer les sorties de systèmes. Leech (1993) rappelle qu'un processus d'annotation comprend plusieurs étapes : écriture et mises à jour du guide d'annotation, entraînement des annotateurs humains, double annotation et phases d'adjudication. Plusieurs solutions existent pour réduire le coût, le temps et améliorer la qualité des annotations : pré-annotation automatique (Dandapat *et al.*, 2009; Fort & Sagot, 2010), apprentissage actif par Ganchev *et al.* (2007) sur du repérage d'entités nommées (REN) et par Voutilainen (2012) et Yimam *et al.* (2014) sur de l'étiquetage en parties du discours, annotation agile (Alex *et al.*, 2010), ou encore propagation contrôlée d'annotations existantes (Grouin, 2016).

Alors que ces solutions préservent le schéma d'annotation, une autre piste de simplification du processus d'annotation consiste à modifier le schéma par fusion de types ou redéfinition. Wilson & Thomas (1997, p. 54) rappellent que les schémas sémantiques relèvent d'un compromis entre le souhait de refléter l'organisation des mots dans l'esprit humain, et la nécessité d'avoir des annotations utiles pour les chercheurs, cette utilité étant guidée par la tâche. Si un schéma d'annotation n'est jamais défini en fonction de possibilités techniques, il est possible de tenir compte des propriétés des données (distribution, ambiguïté, etc.) pour améliorer la qualité des annotations. Un équilibre doit être trouvé entre simplification et possibilité de recouvrer le niveau de détail du schéma d'origine.

Dans cet article, nous évaluons l'impact de la complexité d'un schéma d'annotation sur les performances d'un système de reconnaissance d'entités nommées. Depuis un schéma d'annotation complexe (nombre élevé de catégories, ambiguïtés entre catégories, prise en compte du contexte pour décider d'annoter et déterminer l'étiquette à utiliser, etc.), nous réduisons la complexité en fusionnant des types existants ou en proposant de nouvelles définitions pour les types difficiles à traiter. Nous évaluons les résultats obtenus par des modèles entraînés sur ces versions, et la possibilité de retrouver les types du schéma d'origine après simplification.

2 État de l'art

Travailler sur les schémas d'annotation peut viser un objectif d'interopérabilité (Blache *et al.*, 2010) ou d'analyse de l'impact d'un schéma sur des outils du TAL appliqués sur ces annotations, en particulier pour démontrer l'impact des propriétés du corpus et des différences d'annotations (définitions, nombre de types, portions annotées, coordination d'entités). Plusieurs travaux ont déterminé le seuil au-delà duquel un schéma d'annotation devenait trop complexe.

Sur une tâche d'analyse de dépendances, Mille *et al.* (2012) ont graduellement ajouté des relations grammaticales au schéma d'annotation du Penn Treebank (15, 31, 44 et 60 relations) et ont comparé les performances sur quatre analyseurs de dépendances. Ils observent qu'un schéma d'annotation linguistiquement plus riche n'implique pas forcément une baisse d'exactitude (moins de 0,9) entre 15 et 44 relations alors que les différences sont plus marquées avec le schéma composé de 60 relations.

En reconnaissance d'entités nommées, Shmanina *et al.* (2013) font l'observation inverse. Les auteurs ont comparé les performances de Banner (Leaman & Gonzales, 2008) sur un corpus annoté en suivant deux schémas d'annotations des maladies. Avec une validation croisée 10-plis, l'outil obtient de moins bons résultats sur le corpus annoté avec le schéma d'annotation complexe (F-mesure de 0,7365) qu'avec un schéma d'annotation plus simple (F-mesure de 0,9164).

3 Corpus et méthodes

Nos expériences reposent sur le corpus produit pour la tâche de désidentification de la campagne d'évaluation i2b2/UTHealth 2014 NLP Challenge (Stubbs *et al.*, 2015). Ce corpus comprend 1304 documents cliniques rédigés en anglais (de 2 à 5 par patient), pour un total de 296 patients différents. Les annotations couvrent 7 types principaux (Age, Contact, Date, ID, Location, Name, et Profession) et 25 sous-types (doctor, patient, username pour Name ; street, state, city, zip, hospital, organization pour Location). Ces annotations ont ensuite été remplacées en corpus par des données fictives réalistes.

Nous nous intéressons à deux principaux types d'entités nommées : les noms de personnes en raison de leur complexité sémantique (différence entre nom et prénom, ou entre un médecin et un patient), et les noms de lieu en raison de leur composition d'éléments de types différents, en particulier sur les adresses et noms d'hôpitaux qui sont complexes à traiter pour des systèmes de REN lorsque ces noms font référence à des personnes ou à d'autres lieux.

Nos expériences reposent sur trois versions seulement d'un schéma d'annotation (voir Figure 1) :

— la version d'origine (v1) composée de différences sémantiques (distinction médecin/patient) qui peuvent être ambiguës puisque ces entités sont de type "nom de personne" composées de prénoms et de noms [1]. Cette version se fonde sur des types d'entités reposant sur des définitions humaines, ces types étant conçus pour être directement exploitables en sortie (ici, une différence entre médecins et patients) ;

— une version simplifiée (v2) composées de types d'entités globaux (personnes et lieux) qui font généralement consensus dans la communauté du REN ;

— une version plus régulière (v3) pour laquelle la fonction d'un élément ne détermine pas la catégorie :

  — les prénoms, noms de familles, de villes et d'États sont toujours étiquetés comme 'first', 'last', 'city' et 'state', même s'ils sont utilisés comme composants d'une adresse ou d'un nom d'hôpital *(86 Paris Rd ; 89 James Street ; Cooley Dickinson Hospital ; Johnsonville Family Clinic ; Southwest Texas Medical Center ; Zucker Hillside Hospital)* ;

  — tous les chiffres sont étiquetés 'number' (y compris dans les adresses et codes postaux) ;

  — les initiales de médecins et les codes internes aux hôpitaux *(XD ; MAL60 ; lc855 ; ullmann)* sont rassemblés sous l'étiquette 'code'.

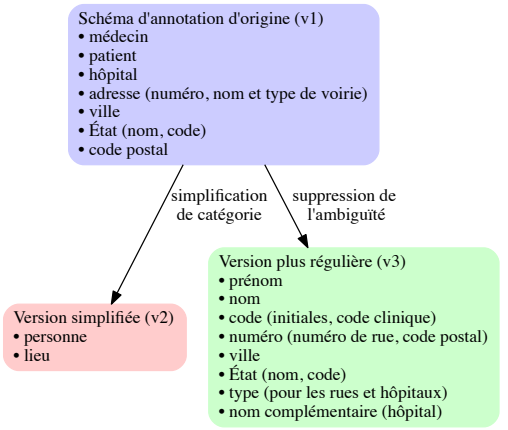


FIGURE 1: Évolution du schéma d'annotation, de la version d'origine (v1) vers les versions simplifiée (v2) et plus régulière (v3)

Type	Entraînement (521 fichiers)			Test (514 fichiers)		
	v1	v2	v3	v1	v2	v3
Person	–	2809	–	–	2791	–
Doctor	1932	–	–	1913	–	–
Patient	879	–	–	879	–	–
First	–	–	1661	–	–	1670
Last	–	–	2663	–	–	2586
Code	–	–	308	–	–	217
Location	–	1695	–	–	1610	–
Hospital	928	–	–	875	–	–
Address	144	–	–	136	–	–
City	259	–	702	264	–	721
State	222	–	243	190	–	205
Zip	139	–	–	140	–	–
Number	–	–	280	–	–	273
Name	–	–	692	–	–	624
Type	–	–	507	–	–	467

TABLE 1: Nombre d'annotations par type dans les corpus d'entraînement et de test pour chaque version du schéma d'annotation

1. Il n'est pas possible de savoir si *Paul Martin* est un médecin ou un patient sans analyser le contexte (sauf en cas de médecin ou de patient connu).

Les trois versions du schéma d'annotation sont présentées sur la phrase 1 pour les noms de personnes et la phrase 2 pour les lieux. Les boîtes extérieures (en rouge) renvoient à la v2, les boîtes intermédiaires (en bleu) à la v1, et les boîtes intérieures (en vert) à la v3.

(1) Ms. person patient first Michelle last Klein was seen in general neurology clinic today following her recent admission for complex migraine. Dr. person doctor first Remigio L. last Allison was present for all salient aspects of the history and physical exam.

(2) Internal Medicine.
loc street number 86 city Paris type Rd
loc city city Washington , loc state state DC loc zip number 20006 .

Afin de produire les versions 2 (simplifiée) et 3 (plus régulière), nous avons converti la version d'origine du schéma d'annotation au moyen des heuristiques suivantes :

— pour produire la v2, nous avons rassemblé les catégories 'Doctor' et 'Patient' sous 'Person', et de manière similaire pour 'Street', 'City', 'State' et 'Zip' sous 'Location' ;

— pour produire la v3, nous avons appliqué des règles définies empiriquement pour réintroduire les prénoms et noms de famille à la place de la distinction 'Patient' et 'Doctor' [2], et scindé les adresses en 'Number', 'Name', et 'Type' [3]. Un contrôle manuel de ces modifications automatiques a été réalisé en deux temps (4h12 pour la première vérification et 1h06 pour la deuxième).

Le tableau 1 présente le nombre total d'annotations par type dans chaque version du schéma d'annotation sur les corpus d'entraînement (521 fichiers) et de test (514 fichiers).

Méthodes Pour chaque version du schéma d'annotation, nous avons identifié les entités nommées grâce à un système d'apprentissage statistique, puis appliqué des règles de conversion pour retrouver les types d'entités nommées de la version d'origine. Le schéma 2 résume les différentes étapes suivies.

Nous avons utilisé l'outil NeuroNER fondé sur des réseaux de neurones récurrents (bi-LSTM) pour identifier les entités nommées. Cet outil prend en entrée un corpus annoté au format BRAT ou ConLL et produit une sortie annotée en entités nommées. Nous avons conservé la configuration d'origine de l'outil et renvoyons à Dernoncourt *et al.* (2017) pour de plus amples détails sur l'architecture du réseau de neurones.

2. Dans les documents traités, le prénom précède toujours le nom (« Michelle Klein ») sauf si le premier élément est suivi d'une virgule, auquel cas le nom précède le prénom (« Glenn, Olivia »). Les règles visent à identifier la frontière entre prénom et nom en fonction du nombre de tokens dans la portion et en tenant compte de la présence d'initiales (« Remigio L. Allison »).

3. Ce découpage repose sur des listes (hôpitaux, États), des déclencheurs (types d'hôpitaux et de voiries) et des règles (éléments numériques).

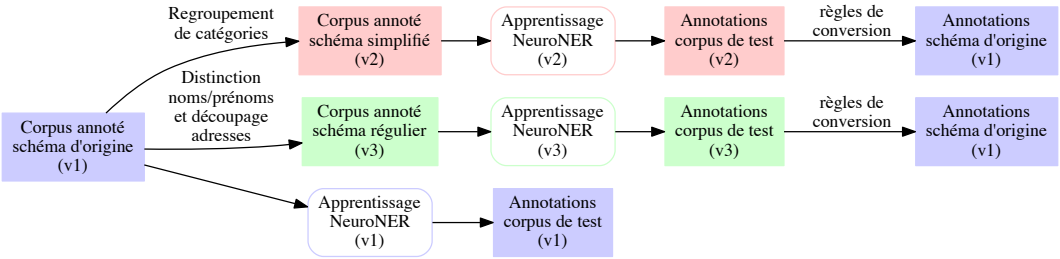


FIGURE 2: Étapes suivies : conversion du schéma d'annotation d'origine en versions simplifiées (v2) et régulière (v3), construction de modèles statistiques pour ces trois versions, application des modèles sur le corpus de test, et conversion des annotations v2 et v3 vers le schéma d'origine

La tokénisation a été réalisée avec l'outil spaCy [4] pour l'anglais, nous avons repris les plongements lexicaux entraînés avec GloVe (Pennington *et al.*, 2014) fournis avec l'outil NeuroNER (200 dimensions pour les plongements lexicaux et 15 dimensions pour les plongements de tokens), l'optimisation est fondée sur un SGD (stochastic gradient descent), et nous avons conservé le seuil d'apprentissage par défaut de 0.005. La construction du modèle a été réalisée en utilisant 8 CPU (<12 minutes par itération).

Nous avons créé des règles de conversion pour retrouver les types d'entités nommées de la version d'origine du schéma d'annotation depuis les sorties produites en versions simplifiée (v2) et régulière (v3). Les règles se fondent sur des déclencheurs et une analyse du contexte. Elles visent uniquement à convertir les types d'EN, sans chercher à identifier de nouvelles entités.

Depuis la version simplifiée (v2), nous avons produit des règles de conversion uniquement pour traiter les cas les plus fréquents (soit une dizaine d'heures de travail). Pour les cas non couverts par nos règles, nous avons utilisé des valeurs par défaut, même si cela engendre une baisse de précision pour les types d'entités utilisés comme valeurs par défaut. Ces valeurs reposent sur les types les plus utilisés en corpus (soit 'Doctor' pour un nom de personne et 'Hospital' pour un nom de lieu). Sur les noms de personnes, nous utilisons les indices de conversion suivants :

— vers 'Doctor' : déclencheurs en contexte gauche *'Att :' 'Attending :' 'CC :' 'Dr' 'Dr.' 'Fellow :' 'Intern :' 'MD :' 'RefMD :'*, droit *'MD' 'M.D.'*, et expressions introductrices *'as per by' 'Dear' 'follow-up X months'* , etc.

— vers 'Patient' : déclencheurs *'Mr' 'Mr.' 'Mrs' 'Mrs.' 'Ms' 'Ms.'* et expressions introductrices *'Patient :' 'Pt :' 'Patient Name :' 'HPI :' 'RE :' 'Impression :' 'to see' 'of seeing' 'I saw' 'w/ sister' 'your patient'*, etc.

Concernant les lieux, notre approche repose sur des règles simples pour retrouver les types d'origines : séquence de 5 chiffres pour 'Zip', liste d'États américains pour 'State', une chaîne de caractères commençant par une majuscule suivie d'un État et d'un code postal pour 'City', la combinaison d'un numéro, d'un nom et d'un type de voirie (*'Avenue' 'Circle' 'Ct' 'Drive' 'Dr' 'Lane' 'Place' 'Road' 'Street' 'STREET' 'St' 'Terrace' 'Way'*) pour 'Street'. Depuis la version régulière (v3), nous avons rassemblé dans la même portion les noms et prénoms des personnes et avons repris les mêmes déclencheurs et phrases introductives que ci-dessus.

4. https://spacy.io

4 Résultats

La figure 3 présente l'évolution de la F-mesure de NeuroNER sur le test, pour chaque version du schéma. Le tableau 2 présente les résultats par catégorie à l'issue de la construction du modèle. L'apprentissage prend fin lorsque les dix dernières itérations n'ont présenté aucune amélioration. Le tableau 3 présente les résultats par application de NeuroNER et des règles de conversion des types d'EN produits par NeuroNER. Puisque les types produits sur la version d'origine du schéma d'annotation (v1) constituent déjà les types d'EN attendus, les résultats sur la v1 sont identiques entre les tableaux 2 et 3. La tableau 4 présente les résultats lorsque la conversion est appliquée sur les entités nommées de référence. Cette évaluation met en évidence la qualité des règles de conversion indépendamment de l'identification des entités nommées par NeuroNER.

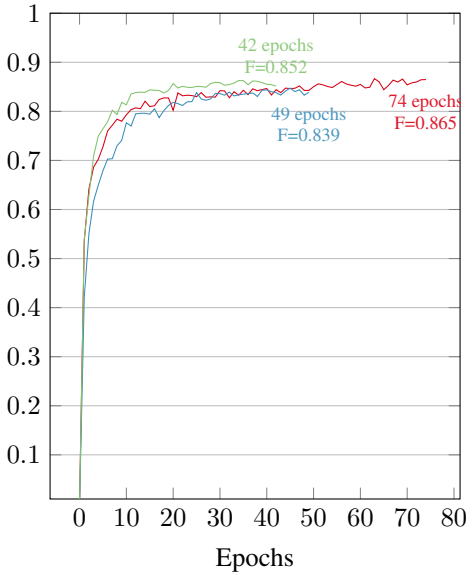


FIGURE 3: Évolution de la F-mesure sur le test par itération pour les versions d'origine (v1), simplifiée (v2) ou régulière (v3)

Type	v1 (origine)			v2 (simple)			v3 (régulier)		
	R	P	F	R	P	F	R	P	F
Pers.	–	–	–	.885	.938	.911	–	–	–
Doc.	.859	.929	.893	–	–	–	–	–	–
Pat.	.846	.863	.854	–	–	–	–	–	–
First	–	–	–	–	–	–	.887	.919	.903
Last	–	–	–	–	–	–	.917	.927	.922
Code	–	–	–	–	–	–	.657	.888	.755
Loc.	–	–	–	.737	.836	.784	–	–	–
Hosp.	.655	.830	.732	–	–	–	–	–	–
Street	.838	.826	.832	–	–	–	–	–	–
City	.633	.788	.702	–	–	–	.633	.761	.691
State	.790	.802	.796	–	–	–	.756	.791	.773
Zip	.914	.928	.921	–	–	–	–	–	–
Num.	–	–	–	–	–	–	.834	.931	.883
Name	–	–	–	–	–	–	.488	.743	.589
Type	–	–	–	–	–	–	.842	.879	.860
TOUS	.800	.881	.839	.831	.902	.865	.818	.888	.852

TABLE 2: Évaluation (rappel, précision, F-mesure) du repérage d'entités nommées selon le schéma d'annotation utilisé pour créer le modèle

5 Discussion

Concernant le repérage d'entités nommées, la figure 3 met en évidence deux conclusions : (i) un schéma d'annotation complexe (v1 avec 7 types) obtient de moins bons résultats que la version simplifiée (v2 avec 2 types) : F=0.839 vs. F=0.865 ; (ii) un schéma plus régulier (v3 avec 8 types) permet à NeuroNER d'obtenir plus rapidement de meilleurs résultats (la courbe verte augmente plus rapidement que les autres). Les conclusions que nous pouvons tirer du tableau 2 sont plus contrastées. Comme attendu, la distinction entre patient et médecin produit de moins bons résultats (F=0.854 and 0.893 respectivement) que celle entre prénom et nom (F=0.903 and 0.922), ce qui prouve sa complexité pour un système statistique. Malgré la réduction de l'ambiguïté dans le schéma

Cat.	v1 (origine)			v2 (simple)			v3 (régulier)		
	R	P	F	R	P	F	R	P	F
Doc.	.859	.929	**.893**	.848	.800	.823	.788	.673	.726
Patient	.846	.863	**.854**	.713	.896	.794	.473	.447	.460
Hosp.	.655	.830	**.732**	.591	.611	.601	.459	.634	.532
Street	.838	.826	**.832**	.596	.853	.701	.588	.370	.455
City	.633	.788	**.702**	.617	.593	.605	.481	.418	.447
State	.790	.802	.796	.795	.853	.823	.816	.856	**.836**
Zip	.914	.928	.921	.900	.984	**.940**	.900	.984	**.940**
TOUS	.800	.881	**.839**	.747	.774	.760	.640	.607	.623

TABLE 3: Rappel, précision, et F-mesure des prédictions de NeuroNER avec application de post-traitement. Les meilleurs résultats sont en gras

v1 (origine)			v2 (simple)			v3 (régulier)		
R	P	F	R	P	F	R	P	F
.859	.929	**.893**	.950	.834	.888	.879	.702	.780
.846	.863	**.854**	.735	.896	.808	.507	.471	.488
.655	.830	.732	.830	.724	**.774**	.753	.747	.750
.838	.826	**.832**	.721	.961	.824	.794	.480	.598
.633	.788	**.702**	.761	.638	.694	.883	.581	.701
.790	.802	.796	.842	.856	.849	.953	.924	**.938**
.914	.928	.921	.943	.971	.957	.957	.971	**.964**
.800	.881	**.839**	.860	.815	.837	.664	.783	.718

TABLE 4: Rappel, précision, et F-mesure des conversions sur les entités nommées de référence. Les meilleurs résultats sont en gras

d'annotation plus régulier, nous obtenons des résultats plus faibles pour les types d'EN utilisés dans plusieurs contextes (F=0.691 vs. 0.702 pour les villes, F=0.773 vs. 0.796 pour les États) par rapport au schéma d'origine. Puisque les systèmes statistiques sont sensibles aux variations contextuelles, une plus grande variété de contextes dans lesquelles apparaissent des EN de ces types a un impact négatif sur les performances du système. De manière non intentionnelle, nous avons remplacé l'ambiguïté de définition par une ambiguïté de contexte. La conversion des types d'EN depuis les deux versions simplifiées se révèle complexe dans la mesure où elle revient à réintroduire de l'ambigüité dans un jeu d'annotations simplifiées.

6 Conclusion

Dans cet article, nous avons comparé l'impact de deux versions d'un schéma d'annotations sur un système de REN, une version simplifiant les types d'EN en deux types consensuels (nom et lieu) et une version proposant des définitions régulières. Dans les deux cas, cette simplification améliore les résultats. Nous avons étudié la possibilité de recouvrer les types d'EN du schéma d'origine à partir des versions simplifiées. Les règles de conversion ne suffisent pas pour retrouver le niveau de détail du schéma d'annotation d'origine. Si l'intérêt d'une simplification est réel, nous estimons que d'autres solutions doivent être identifiées pour retrouver les types d'EN d'origine. Dans le cadre de travaux futurs, nous envisageons de reproduire ces expériences au moyen de CRF de chaîne linéaire, ainsi que l'application de méthodes symboliques. La comparaison des résultats obtenus avec ces approches permettra de vérifier l'impact de la simplification de schémas d'annotation avec d'autres approches.

7 Remerciements

Ce travail a été réalisé dans le cadre du groupe de travail « Sécurité des Données Textuelles » du Labex DigiCosme (projet ANR-11-LABEX-0045-DIGICOSME) financé par l'ANR au travers du programme "Investissement d'Avenir" Idex Paris-Saclay (ANR-11-IDEX-0003-02).

Références

ALEX B., GROVER C., SHEN R. & KABADJOV M. (2010). Agile corpus annotation in practice : an overview of manual and automatic annotation of CVs. In *Proc of LAW*, p. 29–37, Uppsala, Sweden.

BLACHE P., BIGI B., PREVOT L., RAUZY S. & SEINTURIER J. (2010). Annotation schemes, annotation tools and the question of interoperability : from typed feature structures to XML schemas. In *Proc of International Conference on Global Interoperability for Language Resource*, Hong Kong, China.

DANDAPAT S., BISWAS P., CHOUDHURY M. & BALI K. (2009). Complex linguistic annotation – no easy way out ! a case from Bangla and Hindi POS labeling tasks. In *Proc of LAW*, p. 10–18, Suntec, Singapore.

DERNONCOURT F., LEE J. Y. & SZOLOVITS P. (2017). NeuroNER : an easy-to-use program for named-entity recognition based on neural networks. In *Proc of EMNLP*, Copenhagen, Denmark.

FORT K. & SAGOT B. (2010). Influence of pre-annotation on POS-tagged corpus development. In *Proc of LAW*, p. 56–63, Uppsala, Sweden.

GANCHEV K., PEREIRA F., MANDEL M., CARROLL S. & WHITE P. (2007). Semi-automated named entity annotation. In *Proc of LAW*, p. 53–56, Prague, Czech Republic.

GROUIN C. (2016). Controlled propagation of concept annotations in textual corpora. In *Proc of LREC*, Portorož, Slovenia.

LEAMAN R. & GONZALES G. (2008). Banner : an executable survey of advances in biomedical named entity recognition. In *Proc of Pacific Symposium on Biocomputing*, p. 652–63, Hawaii, USA.

LEECH G. (1993). Corpus annotation schemes. *Lit Linguist Computing*, **8**(4), 275–281.

MILLE S., BURGA A., FERRARO G. & WANNER L. (2012). How does the granularity of an annotation scheme influence dependency parsing performance ? In *Proc of COLING, posters*, p. 839–852, Mumbai, India.

PENNINGTON J., SOCHER R. & MANNING C. D. (2014). GloVe : global vectors for word representation. In *Proc of EMNLP*, volume 12, p. 1532–43.

SHMANINA T., ZUKERMAN I., YEPES A. J., CAVEDON L. & VERSPOOR K. (2013). Impact of corpus diversity and complexity on NER performance. In *Proc of Australasian Language Technology Association Wrokshop*, p. 91–95, Brisbane, Australia.

STUBBS A., KOTFILA C. & UZUNER O. (2015). Automated systems for the de-identification of longitudinal clinical narratives : Overview of 2014 i2b2/UTHealth shared task track 1. *J Biomed Inform*, **58**, S11–S19.

VOUTILAINEN A. (2012). Improving corpus annotation productivity : a method and experiment with interactive tagging. In *Proc of LREC*, p. 2097–2102, Istanbul, Turkey.

WILSON A. & THOMAS J. (1997). Semantic annotation. In R. GARSIDE, G. LEECH & T. MCENERY, Eds., *Corpus Annotation*, chapter 4, p. 53–65. Routledge.

YIMAM S. M., DE CASTILHO R. E., GUREVYCH I. & BIEMANN C. (2014). Automatic annotation suggestions and custom annotation layers in WebAnno. In *Proc of ACL, System Demonstrations*, p. 91–96, Baltimore, MA.

Apprentissage déséquilibré pour la détection des signaux de l'implication durable dans les conversations en parfumerie

Yizhe WANG[1] Damien NOUVEL[2] Marguerite LEENHARDT [1] Gaël PATIN[1]
(1) XiKO, 87 rue Gabriel Péri, Paris, France
(2) ERTIM, 2 rue de Lille, Paris, France
wangyizhe0201@gmail.fr, damien.nouvel@inalco.fr,
marguerite.leenhardt@xiko.fr, gael.patin@xiko.fr

RÉSUMÉ

Une simple détection d'opinions positives ou négatives ne satisfait plus les chercheurs et les entreprises. Le monde des affaires est à la recherche d'un «aperçu des affaires». Beaucoup de méthodes peuvent être utilisées pour traiter le problème. Cependant, leurs performances, lorsque les classes ne sont pas équilibrées, peuvent être dégradées. Notre travail se concentre sur l'étude des techniques visant à traiter les données déséquilibrées en parfumerie. Cinq méthodes ont été comparées : Smote, Adasyn, Tomek links, Smote-TL et la modification du poids des classe. L'algorithme d'apprentissage choisi est le SVM et l'évaluation est réalisée par le calcul des scores de précision, de rappel et de f-mesure. Selon les résultats expérimentaux, la méthode en ajustant le poids sur des coût d'erreurs avec SVM, nous permet d'obtenir notre meilleure F-mesure.

ABSTRACT

Automatic detection of positive enduring involvement signals in fragrance products reviews

Opinion mining have been widely studied in natural language processing. Nevertheless, in recent years, a simple detection of positive or negative opinion can no longer satisfy researchers and companies. The business world is looking for "business insights". A lot of machine learning algorithms can be used to deal with the problem. However, their performance on imbalanced data can be degraded. In this paper, we focus on the study of techniques aimed at treating imbalanced data in the perfume sector. Five methods were compared : Smote, Adasyn, Tomek links, Smote-TL and changing weight of the class. The selected standard classifier is SVM and precision, recall and F-mesure scores are calculated for evaluation. According to experimental results, the method by adjusting the cost weight allows us to get our best F-measure.

MOTS-CLÉS : fouille d'opinions, classification asymétrique, SVM, ré-échantillonnage, apprentissage sensible aux coûts.

KEYWORDS: opining mining, imbalanced classification, SVM, resampling, cost sensitive learning.

1 Introduction

Aujourd'hui, les entreprises attendent plus qu'une simple détection d'opinion positive ou négative, mais des appréciations plus fines comme l'intention d'achat, la préférence pour les produits, la fidélité à la marque, y compris l'implication durable qui aide les experts en marketing à trouver l'explication du comportement de rachat au niveau individuel. Bien que la détection de ces signaux

liés au marketing ne soit pas encore beaucoup étudiée en fouille d'opinions, de nombreux algorithmes d'apprentissage automatique peuvent être utilisés pour ce faire : réseaux de neurones, arbres de décision, machines à vecteurs de support, etc.

Dans notre travail, nous considérons l'implication durable comme un signal binaire, présent ou absent. Nous cherchons à résoudre le problème sur un corpus déséquilibré dédié à la parfumerie, en comparant la performance des différentes techniques. Les machines à vecteur de support sont utilisées comme algorithme de classification. Notre hypothèse est qu'un algorithme d'apprentissage sensible aux coûts devrait surpasser les méthodes de ré-échantillonnage.

Nous présenterons d'abord le contexte général et notre positionnement. Ensuite, nous étudierons les techniques que nous avons choisies pour résoudre nos problèmes sur la classification asymétrique. La partie suivante est consacrée à la présentation de notre corpus et nos règles d'annotation. Les résultats seront montrés dans la partie suivante et nous terminerons par une discussion suivie par la conclusion.

2 Contexte général

2.1 Notion d'implication durable

L'implication est une notion issue de la psychologie qui interprète l'implication d'un individu en étudiant sa relation avec une autre personne, une cible ou un sujet. Bien qu'elle soit étudiée depuis plus de 30 ans en marketing, ce concept reste difficile à appréhender en raison de son interdépendance avec des concepts variés d'autres disciplines. Néanmoins, en marketing, il fait consensus que cette implication est une variable intrinsèque au niveau individuel, assimilée à l'attachement personnel aux objectifs ou aux événements (Abdolvand & Nikfar, 2012). Il y a trois types de classification de l'implication en marketing, dont le classement par nature qui a été proposée par Rothschild en 1975 et est acceptée assez largement : l'implication durable (EI) et l'implication situationnelle (SI).

Contrairement à la SI qui est liée à la situation temporaire du consommateur à l'égard du produit, l'EI est considérée comme un état stable du consommateur auprès d'un produit (Houston, 1978). Elle représente un niveau d'intérêt ou d'attachement d'un individu envers un produit à long terme. D'après Valette-Florence (Valette-Florence, 1989), l'EI se réfère à la fois à l'expérience ou à la connaissance antérieure du produit et aux valeurs auxquelles adhèrent l'individu. C'est-à-dire que l'EI sera positive pour une personne qui a testé ou utilisé un produit et qui a envie de continuer à l'utiliser pour longtemps ou qui lui porte une admiration intense.

2.2 Travaux précédents

Deux types de méthodes sont souvent utilisées en fouille d'opinions : l'approche symbolique basée sur le lexique et l'approche statistique en utilisant l'apprentissage automatique. L'approche basée sur le lexique utilise généralement une liste de mots ou d'expressions qui portent sur les opinions ou les sentiments des humains. On peut aussi utiliser les listes des termes contenant des sentiments qui existent déjà, comme SentiWordNet, SenticNet et HowNet. Les méthodes statistiques procèdent par classification, l'étape essentielle étant de choisir les caractéristiques, lexicales, syntaxique ou sémantiques, afin de représenter les messages.

Parmi les méthodes possibles pour sélectionner les caractéristiques, TF-IDF est largement utilisé. (Su *et al.*, 2014) ont essayé d'utiliser Word2Vec puis un classifieur SVM pour des commentaires, ils obtiennent une exactitude de plus de 90%. (Le & Zuidema, 2015) propose une méthode d'analyse de sentiments en utilisant un modèle de réseaux de neurones LSTM. (Hassan, 2017), avec une méthode similaire, montre que l'utilisation de vecteurs de mots obtenus à partir d'un modèle de réseaux de non supervisé de plongements de mots comme caractéristiques d'un système RNN-LSTM peut augmenter la performance du système de fouille d'opinions.

Le problème que nous traitons dans cet article est que le nombre d'instances dans les classes est presque toujours déséquilibré. Les travaux de (Leenhardt & Patin, 2014) et Li *et al.* (2011) se basent sur une méthode d'apprentissage semi-supervisé en modifiant la technique du sous-échantillonnage pour la classification asymétrique de sentiment. Au lieu de faire le ré-échantillonnage, Krawczyk *et al.* (2014) ont créé un ensemble efficace d'arbres de décision sensibles aux coûts pour la classification asymétrique. Une nouvelle approche pour la classification des données déséquilibrées a été proposée par Zhang *et al.* (2017), qui montre les bons résultats par minimisation du coût.

3 Méthodes

La performance des algorithmes baisse en face des données déséquilibrées. Ceci peut être expliqué par le fait que ces algorithmes cherchent à minimiser le taux d'erreur, et ignorent la différence entre les différents types de classification erronée. En particulier, ils supposent implicitement que toutes les erreurs de classification représentent le même coût lors de l'apprentissage du modèle (Ganganwar, 2012), ce qui n'est pas toujours optimal. Deux types de méthodes sont à considérer : celles qui modifient les données d'apprentissage, et celles qui adaptent les algorithmes.

3.1 Approches au niveau des données

Ces approches, basées sur le ré-échantillonnage, cherchent à augmenter la fréquence de la classe minoritaire ou à diminuer celle de la classe majoritaire. Ceci est fait afin d'obtenir approximativement le même nombre d'instances pour les classes. Nous avons testé les algorithmes présentés ci-dessous.

— **Smote** : Cette méthode de sur-échantillonnage se concentre sur la classe minoritaire, qui est augmentée en créant des exemples «synthétiques». C'est l'un des algorithmes les plus utilisés pour améliorer la performance de classifieurs appliqués sur les données déséquilibrées. L'agorithme fournit un ensemble de règles simples pour générer de nouvelles données «synthétisées». Bien que chaque nouvelle donnée synthétique soit construite à partir de ses parents (la donnée choisie et l'un de ses voisins les plus proches), la donnée générée n'est jamais un double exact de l'un de ses parents.

— **Adasyn** (Adaptive synthetic sampling) : Cet algorithme a été proposé en 2008 par He *et al.* (2008). L'idée essentielle est d'utiliser une distribution pondérée pour différents groupes de la classe minoritaire en fonction de leur niveau de difficulté d'apprentissage. Plus les données sont difficiles à apprendre, plus le nombre de données synthétiques générées va être important. L'approche améliore l'apprentissage par rapport aux distributions de données de deux façons : elle réduit le biais introduit par le déséquilibre des classes et déplace de façon adaptative la limite de classification à l'égard des exemples difficiles à apprendre.

— **Tomek links** Ces liens sont des paires de données qui sont les plus proches autour de la ligne de séparation. Cela signifie que ce sont les données qui vont être les plus problématiques pour la plupart des algorithmes de classification. En supprimant ces paires de données, la séparation entre les deux classes sera élargie, de sorte que notre l'algorithme pourra faire moins d'erreurs.
— **Smote-TL** : La méthode hybride combine les approches de sur-échantillonnage et celles de sous-échantillonnage en éliminant des données dans la classe majoritaire et ajoutant des données dans la classe minoritaire afin de rééquilibrer la distribution des classes (Santoso *et al.*, 2017). L'approche Smote-TL est la combinaison des algorithmes Smote et Tomek Links. Elle a d'abord été utilisée pour améliorer la classification des exemples sur le problème de l'annotation des protéines en bioinformatique (Batista *et al.*, 2003).

3.2 Approche algorithmique

Parmi les solutions algorithmiques (qui ne modifient pas les données) possibles, en tenant compte de l'effort pour la réalisation et de l'implémentation, nous avons choisi l'apprentissage sensible aux coûts qui tient compte des coûts associés aux exemples mal classés(Ting, 2002) selon leur proportion dans les données. Plus concrètement, cette méthode cible le problème d'apprentissage déséquilibré en utilisant des matrices de coûts qui décrivent les coûts de classification erronée. Pour une classification binaire, la matrice de coûts se concentre sur les faux positifs et les faux négatifs. Il n'y a aucun ajustement de poids associé aux vrais positifs et vrais négatifs car ils sont correctement identifiés.

Dans l'intention de trouver un meilleur poids pour notre corpus, on a essayé le poids de 1 fois à 15 fois à l'originel et on a pris finalement le poids 2, qui donne le meilleur résultat en terme de f-mesure.

4 Expérimentations

Pour nos expériences, nous avons sélectionné le SVM comme classifieur, puisque cet algorithme est réputé être plus précis sur des données modérément déséquilibrées. Les vecteurs de support sont utilisés pour la classification, ainsi de nombreux échantillons majoritaires éloignés de l'hyperplan de séparation peuvent être supprimés sans affecter la classification (Akbani *et al.*, 2004). Cependant, ce classifieur peut être sensible à un fort déséquilibre entre les classes.

Dans nos expériences, les scores rappportés en rappel, précision et f-mesure ne concernent que la classe ciblée. La sélection et l'optimisation du modèle sont faites selon la f-mesure.

4.1 Corpus et règles d'annotation

Notre corpus est constitué de commentaires en français sur le site d'avis beaute-test.com [1], spécialisé dans les produits de beauté. Nous avons extrait 20K verbatims sur le site et en avons sélectionné aléatoirement 9180 pour construire notre corpus. La longueur du commentaire varie largement : de 1 mot à 2514 mots. L'implication durable est une relation consommateur-objet stable basée sur les besoins inhérents du consommateur. Lorsque le comportement du consommateur tend à un objectif

1. http://www.beaute-test.com

à long terme (Ogbeide, 2014) ou reflète les sentiments durables à l'égard d'un produit ou d'une catégorie (Sirgy *et al.*, 2014). Ainsi, après des discussions avec les experts en marketing, nous avons considéré trois types d'expressions comme des signaux dénotant l'implication durable :

— l'expression de l'intention d'une utilisation prolongée :
« Je porte ce parfum depuis plus de 6 ans et je n'en demords toujours pas...j'adore »
— l'expression du rachat :
« J'ai toujours un flacon chez moi ! ! ! ! C'est mon parfum préféré que je rachèterai encore et encore ..... ! ! ! ! ! ! »
— l'expression d'attachement très forte au niveau de l'adoption :
« Une fragance fruitée subtile et enivrante je recommande ce parfum vous ne serez pas déçue l'essayer une fois c'est l'adopter ! ! ! »

La figure 1 montre des exemples d'expressions pour repérer les signaux demandés.

Expressions de l'intention d'une utilisation prolongée	**Expressions de rachat**	**Expression de l'adoptation**
ne change plus	acheterai encore	adopté
ai toujours un flacon	rachèterais	adoptation
reviens toujours	acheter à nouveau	adopter
c'est mon 4 eme flacon	reprendrai	
plusieurs flacons	y retourner	
encore fidèle		
3 fois que je l'achète		
ne peux m'en passer		
difficilement l'abandonner		
depuis bientôt 3 ans		
ne se quitte plus		
je ne le lâche plus		

FIGURE 1 – Exemples d'expressions contenant les signaux demandés

Des règles sont utilisées pour détecter l'EI. Les signaux positifs de EIs ne sont pas toujours explicites dans les avis publiés. D'ailleurs, toutes les phrases contenant les expressions qui ont les mêmes sens que les expressions prédéfinies par nos règles d'annotation ne comportent pas nécessairement les signaux à détecter. Par exemple, le verbatim « je l'ai utilisé pendant longtemps mais je l'aime plus » pourrait être détecté à tort comme positif. Si l'expression d'affection très forte au niveau de l'adoption est choisie comme un règle d'annotation au lieu de l'ensemble de concepts contenant le sentiment d'admiration, c'est parce qu'il faut éviter des ambiguïtés et que c'est mieux d'avoir une frontière plus claire avec d'autres notions en marketing, comme la préférence. De ce fait, les avis de produits avec un sentiment positif assez fort comme « Je suis tomber amoureuse de ce parfum pour les filles qui aime les notes sucrés je conseil vivement » et « Un de mes parfums préférés ! ! ! » sont annotés en

négatifs. Par ailleurs, tous les échantillons ont été annotés par une personne.

Avant d'entrer dans l'apprentissage, la tokenization et la suppression des mots vides ont été faites à l'aide de la librairie NLTK. Parmi toutes les méthodes d'extraction des caractéristiques implémentées, nous avons choisi le modèle TF-IDF.

4.2 Méthode d'évaluation

Pour faire face au problème généré par le déséquilibre entre les classes et obtenir un modèle de classification asymétrique optimal, la sensibilité et la spécificité sont habituellement adoptées pour surveiller respectivement la performance de classification sur deux classes. Cependant, parfois, on s'intéresse à la capacité de détecter efficacement une seule classe. Pour de tels problèmes, une autre paire de mesures, la précision et le rappel, est souvent adoptée. La f-mesure est aussi souvent utilisée pour intégrer la précision et le rappel dans une seule mesure pour la commodité de l'évaluation.

Notez que tous nos scores (rappel, précision et f-mesure) sont seulement ceux de la classe ciblée.

Dans nos expériences, la sélection et l'optimisation du modèle sont faites en condition de la f-mesure. Cependant, il faut savoir que dans l'application métier, c'est souvent l'objective métier ou la demande du client qui décide du choix de modèle. Par exemple, dans le travail de filtration du spam (la classe positive est spam), la sélection et l'optimisation du modèle sont souvent faites selon la précision. Parce que les faux négatifs (le spam va dans la boîte de réception) sont plus acceptables que les faux positifs.

4.3 Résultats

Le tableau 4.3 montre le changement du nombre de données après avoir utilisé les quatres techniques aux niveaux des données :

	Originel	Adasyn	TL	Smote	Smote+TL
nombre total d'échantillons	9180	13231	7243	13118	13114
classe minoritaire	907	6672	727	6559	6557

TABLE 1 – Changement du nombre d'échantillons

Le tableau 4.3 présente les résultats obtenus par utilisation du SVM sur les données et selon les différentes méthodes de prise en compte du déséquilibre des classes. Une optimisation supplémentaire du SVM a été faite en faisant varier les hyper-paramètres selon une méthode aléatoire, comme proposé par Bergstra & Bengio (2012).

4.4 Discussion

Nous voyons dans les résultats que Tomek Links fonctionne assez bien sur notre corpus. Cependant, en utilisant une méthode de sous-échantillonnage, des informations importantes risquent d'être éliminées en même temps. Il y a deux types de classification asymétrique : le déséquilibre relatif et la rareté absolue (Weiss, 2004). Notre classe minoritaire contient 907 exemples c'est un cas de déséquilibre

	Précision	Rappel	F-mesure
SVM	59.75%	62.50%	61.09%
SVM+opt	58.56%	67.76%	63%
SVM+Smote	50.41%	81.85%	62.31%
SVM+S-TL	53.42%	76.79%	63.07%
SVM+Adasyn	39.02%	**84.21%**	53.33%
SVM+TL	**66.67%**	60.53%	63.45%
SVM+couts	63.20%	67.76%	65.40%
SVM+couts+opt	60.20%	77.63%	**67.82%**

TABLE 2 – Tableau de résultats

relatif. De ce fait, après avoir supprimé des données de la classe majoritaire et avoir des classes assez équilibrées, la performance du SVM serait aussi améliorée.

En comparant l'efficacité de Tomek Links avec celui ajustant le poids de la classe minoritaire, on conclut que le dernier est plus performant sur notre corpus, puisqu'il nous permet d'obtenir notre meilleure f-mesure. C'est un désavantage connu de l'utilisation des méthodes de ré-échantillonnage L'inconvénient du sous-échantillonnage est qu'il provoque l'élimination des données potentiellement utiles. Et l'inconvénient essentiel du sur-échantillonnage est qu'en faisant des copies des exemples existants, un phénomène de sur-apprentissage risque de survenir (Weiss *et al.*, 2007). Un deuxième inconvénient du sur-échantillonnage est qu'il augmente le nombre d'exemples d'apprentissage, augmentant ainsi le temps d'apprentissage, ce qui est visible aussi dans notre expérience.

Pour nos expériences, l'apprentissage par méthode sensible aux coût est celle qui fournit les meilleurs résultats. Cependant, il est difficile de généraliser. L'expérience de Weiss *et al.* (2007) montre que l'algorithme d'apprentissage sensible aux coûts surpasse systématiquement les méthodes de ré-échantillonnage quand on se concentre exclusivement sur des ensembles de données comportant plus de 10K exemples, tandis que la technique de sur-échantillonnage semble être la meilleure méthode pour les petits ensembles de données.

Pour des travaux futurs, nous envisageons de mettre en place d'autres méthodes qui donnent de bons résultats pour la classification des classes déséquilibrées, comme la méthode *Random over-sampling*, une méthode simple mais compétitive par rapport aux autres techniques de sur-échantillonnage plus complexes (Batista *et al.*, 2004).

5 Conclusion

Dans cet article, nous avons évalué 5 algorithmes souvent utilisés en classification asymétrique afin de détecter les signaux positifs de l'implication durable dans les avis des consommateurs en parfumerie. L'algorithme Tomek Links donne les meilleurs résultats en précision, Adasyn pour le rappel. Globalement, l'approche en utilisant l'algorithme sensible aux coûts est la meilleure méthode avec une F-mesure de 67.82%. Si les méthodes de sur-échantillonnage ou de sous-échantillonnage peuvent être intéressantes pour des cas particuliers, l'approche algorithmique est celle qui apporte les meilleurs résultats.

Références

ABDOLVAND M. & NIKFAR F. (2012). Investigation of the relationship between product involvement and brand commitment.

AKBANI R., KWEK S. & JAPKOWICZ N. (2004). Applying support vector machines to imbalanced datasets. *Machine learning : ECML 2004*, p. 39–50.

BATISTA G. E., BAZZAN A. L. & MONARD M. C. (2003). Balancing training data for automated annotation of keywords : a case study. In *WOB*, p. 10–18.

BATISTA G. E., PRATI R. C. & MONARD M. C. (2004). A study of the behavior of several methods for balancing machine learning training data. *ACM Sigkdd Explorations Newsletter*, **6**(1), 20–29.

BERGSTRA J. & BENGIO Y. (2012). Random search for hyper-parameter optimization. *Journal of Machine Learning Research*, **13**(Feb), 281–305.

GANGANWAR V. (2012). An overview of classification algorithms for imbalanced datasets. *International Journal of Emerging Technology and Advanced Engineering*, **2**(4), 42–47.

HASSAN A. (2017). Sentiment analysis with recurrent neural network and unsupervised neural language model.

HE H., BAI Y., GARCIA E. A. & LI S. (2008). Adasyn : Adaptive synthetic sampling approach for imbalanced learning. In *Neural Networks, 2008. IJCNN 2008.(IEEE World Congress on Computational Intelligence). IEEE International Joint Conference on*, p. 1322–1328 : IEEE.

HOUSTON M. J. (1978). Conceptual and methodological perspectives on involvement. *Research frontiers in marketing : Dialogues and directions*, p. 184–187.

KRAWCZYK B., WOŹNIAK M. & SCHAEFER G. (2014). Cost-sensitive decision tree ensembles for effective imbalanced classification. *Applied Soft Computing*, **14**, 554–562.

LE P. & ZUIDEMA W. (2015). Compositional distributional semantics with long short term memory. *arXiv preprint arXiv :1503.02510*.

LEENHARDT M. & PATIN G. (2014). Détecter les intentions d'achat dans les forums de discussion du domaine automobile : une approche robuste à l?épreuve des expressions linguistiques peu répandues.

LI S., WANG Z., ZHOU G. & LEE S. Y. M. (2011). Semi-supervised learning for imbalanced sentiment classification. In *IJCAI proceedings-international joint conference on artificial intelligence*, volume 22, p. 1826.

OGBEIDE O. A. (2014). Knowing your customers to serve them better : Enduring involvement approach. *Global Research Journal of Business Management*, **2**(2), 5–14.

SANTOSO B., WIJAYANTO H., NOTODIPUTRO K. & SARTONO B. (2017). Synthetic over sampling methods for handling class imbalanced problems : A review. In *IOP Conference Series : Earth and Environmental Science*, volume 58, p. 012031 : IOP Publishing.

SIRGY J., RAHTZ D. & DIAS L. (2014). Consumer behavior today. *Irvington, NY : Flatworld Knowledge Publishers*.

SU Z., XU H., ZHANG D. & XU Y. (2014). Chinese sentiment classification using a neural network tool ?word2vec. In *Multisensor Fusion and Information Integration for Intelligent Systems (MFI), 2014 International Conference on*, p. 1–6 : IEEE.

TING K. M. (2002). An instance-weighting method to induce cost-sensitive trees. *IEEE Transactions on Knowledge and Data Engineering*, **14**(3), 659–665.

VALETTE-FLORENCE P. (1989). Conceptualisation et mesure de l'implication. *Recherche et Applications en Marketing (French Edition)*, **4**(1), 57–78.

WEISS G. M. (2004). Mining with rarity : a unifying framework. *ACM Sigkdd Explorations Newsletter*, **6**(1), 7–19.

WEISS G. M., MCCARTHY K. & ZABAR B. (2007). Cost-sensitive learning vs. sampling : Which is best for handling unbalanced classes with unequal error costs ? *DMIN*, **7**, 35–41.

ZHANG C., WANG G., ZHOU Y. & JIANG J. (2017). A new approach for imbalanced data classification based on minimize loss learning. In *Data Science in Cyberspace (DSC), 2017 IEEE Second International Conference on*, p. 82–87 : IEEE.

A comparative study of word embeddings and other features for lexical complexity detection in French

Aina Garí[1] Marianna Apidianaki[1, 2] Alexandre Allauzen[1]
(1) LIMSI, CNRS, Univ. Paris Sud, Université Paris-Saclay, 91403 Orsay
(2) Computer and Information Science Department, University of Pennsylvania
aina.gari@limsi.fr, marianna@limsi.fr, allauzen@limsi.fr

RÉSUMÉ

Etude comparative de plongements lexicaux et autres traits pour la détection de la complexité lexicale en français

Détecter la complexité lexicale est une étape importante pour la simplification automatique de textes, servant lors de l'identification des éléments lexicaux à substituer. Dans ce travail, nous explorons l'utilité des plongements lexicaux pour mesurer la complexité de mots en français, en les combinant avec d'autres traits reconnus comme étant utiles pour cette tâche. Nos résultats sur une tâche d'ordonnancement de synonymes selon leur complexité montrent que les plongements seuls donnent de meilleurs résultats que nombreux autres traits, bien que leur performance reste inférieure à celle de systèmes basés sur la fréquence pour cette langue.

ABSTRACT

Lexical complexity detection is an important step for automatic text simplification which serves to make informed lexical substitutions. In this study, we experiment with word embeddings for measuring the complexity of French words and combine them with other features that have been shown to be well-suited for complexity prediction. Our results on a synonym ranking task show that embeddings perform better than other features in isolation, but do not outperform frequency-based systems in this language.

MOTS-CLÉS : Complexité lexicale, lisibilité, ordonnancement de synonymes, plongements lexicaux.

KEYWORDS: Lexical complexity, readability, synonym ranking, word embeddings.

1 Introduction

Complex word identification (CWI) is an important step in text simplification systems which aim at modifying the complexity level of texts to make them more accessible to readers with reading difficulties. There are many factors that can make a text complex or difficult, at the syntactic level (e.g. texts using passive voice), but also at the pragmatic and lexical levels (e.g. texts with a high number of domain-specific or rare words). Lexical complexity is the property of words that may pose comprehension difficulties to some readers, and which would not be part of the vocabulary of young children or low-proficiency foreign language learners. Complex words have a high impact on the readability of a text. Arya *et al.* (2011) showed that lexical complexity affected comprehension of elementary science texts by children, and manual lexical simplification has been shown to improve

understanding and reading speed in dyslexics (Rello *et al.*, 2013). CWI can serve to identify the lexical elements in a text that need to be substituted by simpler synonyms or paraphrases for the meaning of the text to become more accessible.

We carry out a comparative study of different types of systems for measuring the complexity of French words. We experiment with features that have been identified as good indicators of complexity in the literature and word embeddings which, although successful in numerous semantics-related tasks, have not yet been applied to lexical complexity prediction. We assume that complex words appear in complex contexts, and simple words occur in texts that are easier to understand. We consider lexical complexity to be a property of words that is reflected in their context of use. Word embeddings encode rich contextual information, so we expect them to be particularly useful for complexity detection.

We train our systems on two French lexical resources that encode the distribution of words across different levels of difficulty : Manulex, where categories correspond to school levels (Lété *et al.*, 2004), and the FLElex database where words are categorized into second language proficiency levels (François *et al.*, 2014). We evaluate our systems on lists of synonyms that have been manually ordered by complexity.

2 Related Work

Due to the importance of lexical complexity detection for simplification, there have been numerous approaches to automatically predicting complexity. The task is often regarded as a classification problem into two or more levels of difficulty. Shardlow (2013) performs binary classification of words into simple or complex by means of a Support Vector Machine (SVM) classifier based on features such as frequency, number of syllables and senses. Gala *et al.* (2014) classify French words into three or six complexity levels, which indicate the distribution of words in the three school levels and six second language proficiency levels found in Manulex and FLElex, respectively. Gala *et al.*'s (2014) systems achieve accuracies close to 63% and 43% when trained and evaluated on the two resources. Interestingly, both Shardlow's (2013) and Gala *et al.*'s (2014) systems obtain results that are close to those of baseline models based on frequency only, and show that complex words occur more rarely in texts.

Another approach proposed by François *et al.* (2016) consists in ordering synonyms by complexity, creating thus a ranking instead of a classification. Their system is trained on pairs of French synonyms based on a complexity score derived from Manulex, and uses 21 features. We use in this study three of their best performing features : number of characters, number of phonemes and frequency.

Other works have focused on creating readability lexicons. Kidwell *et al.* (2009) develop a model that infers the age of aquisition of a word from its presence in graded texts and use it to assess text readability. Brooke *et al.* (2012) build a readability lexicon based, among others, on word co-occurrence information, which is the closest feature to the word embeddings we use in our experiments.

In the SemEval 2012 shared task on lexical simplification (Specia *et al.*, 2012), where systems had to choose a simpler synonym for words in context, the frequency baseline was very powerful and was only beaten by one system using psycholinguistic features including concreteness, imageability, familiarity and age of acquisition (Jauhar & Specia, 2012). When investigating the characteristics of lexical complexity for Spanish, Drndarević & Saggion (2012) conclude that combining frequency

Manulex

Class	Original	Balanced
1	9,384	9,384
2	8,040	8,040
3	19,421	9,000
Total	36,845	**26,424**

FLElex

Class	Original	Balanced
1	3,617	2,000
2	2,470	2,000
3	3,661	2,000
4	1,178	1,178
5	1,561	1,561
6	455	455
Total	12,942	**9,194**

TABLE 1 – Number of words available in the original Manulex and FLElex databases and in the balanced datasets. Words are classified per first class of appearance in each resource.

and word length is the best approach for choosing a substitute for a complex word.

3 Data and Resources

We train and evaluate our systems on two large lexical resources for French, **Manulex** and **FLElex**.

Manulex (Lété *et al.*, 2004) is a lexical database with word frequency information calculated from 54 French textbooks used in three school levels, with a total of 1.9 million word tokens. The levels correspond to ages of 6 (CP), 7 (CE1) and 8 to 11 (CE2-CM2) years old. We use its non-lemmatized version, which contains 48,886 word forms, and keep only unigrams with open parts of speech (nouns, adjectives, verbs and adverbs), which reduces it to 39,839 words.

FLElex (François *et al.*, 2014) is a graded lexicon based on textbooks and simplified readers for learners of French as a foreign language which contains a total of 777,000 tokens. This resource is organized in 6 levels corresponding to the proficiency levels of the CEFR scale (Conseil de l'Europe, 2001), ranging from A1 to C2. The resource contains 14,236 lemmas from which we retain 13,250 after filtering for part of speech and multi-words, as for Manulex.

We perform feature extraction and remove words for which no word embedding is available, which leaves us with 36,845 and 12,942 words in each dataset. Since words might occur in different levels, we assign each word in its first level of appearance, indicating its moment of acquisition, and we observe that the distribution of words is biased towards the third level in Manulex and the three first levels in FLElex. We believe this is because the third level in Manulex groups more ages than the first two, and in the case of FLElex the bias probably reflects the need of constructing a substantial lexicon during the first stages of learning. In order to learn from a more balanced dataset, we randomly remove words from the majority classes, obtaining a total of 26,424 words for Manulex and 9,194 for FLElex. The contents of the original and balanced resources are given in Table 1. 95% of the remaining data of each database is used for training, and 5% is used during the development stage to assess the correlation of the system's predictions with respect to the original classes the words belong to. To better estimate the complexity prediction capacity of the models on unseen data, we also remove from the training data words present in the synonym lists used for evaluation.

For evaluation, we use sets of synonyms manually ranked for complexity and made available by François *et al.* (2016). One example is *osseux* → *squelettique* → *maigre*, where words are ordered from more complex to simpler. Forty annotators were asked to order the synonyms according to their reading and comprehension difficulty without any given context. The resource consists of 36 groups

of synonyms and 134 lexical units, with a few words being present in more than one group, ranked according to the majority ranking amongst all annotators. The reported Krippendorff's α of 0.399 reflects the difficulty of this task even for humans. Because of the unavailability of word embeddings for multiword expressions and for some synonyms present in the dataset, [1] we remove 13 words and 3 groups of synonyms (for which there was none or only one synonym left after removing the words), which results in a dataset of 33 sets of synonyms and 121 words. 5 of these sets are used as a development set, which results in a test set of 28 synsets and 104 words.

We extract frequency and number of phonemes of lexical items to be used by our models from **Lexique3** (New *et al.*, 2001), a large resource with various kinds of lexical information (phonetic transcription, syllabic structure, number of morphemes, etc.) for 142,728 French words.

4 Model description

We view complexity as a continuous, rather than categorical, notion with highly complex/simple words situated at the two extremes of the spectrum. In order to induce a continuous representation from the three Manulex levels and the six FLElex levels, we implement a simple feed-forward neural network that has as learning objective a number corresponding to the first level of appearance of a word in the corresponding resource. This serves as an indication of the age level or moment of acquisition of the word. The network takes as input features that represent the target word and outputs a real number that indicates its complexity. This way, we obtain a continuous representation of complexity from categorical data, which allows for a more fine-grained ranking than the one found in the original resources.

We experiment with different feature combinations, number of layers and layer sizes on a development set, and choose the best configuration for each feature combination for evaluation (see Section 5). We use three different architectures : one with no hidden layer, similar to logistic regression (no HL) ; one with a hidden layer of size 100 (1 HL) ; and one with two hidden layers of sizes 150 and 100 (2 HL). In all settings, rectified linear unit (ReLU) is used as an activation function and iterations are limited to 100.

For our experiments, we use the following lexical features which have been shown to be useful for predicting complexity in past work (Gala *et al.*, 2014) :

— **Number of characters**. The hypothesis is that longer words tend to be more complex, and shorter words tend to be simpler.
— **Number of phonemes**. This information is obtained from *Lexique3*, when available. For words not present in *Lexique3*, following Gala *et al.* (2014), we extract the information using eSpeak, an open source speech synthesizer. [2] The hypothesis is the same as for word length measured in terms of characters. We would expect the number of phonemes to be a better indication of word length for French given the abundance of many-to-one grapheme-phoneme correspondences.
— Log **frequency** in a corpus of film subtitles, as encoded in Lexique3. Frequencies in Lexique3 were calculated by averaging the frequency of appearance of a word per million counts in

1. The lexical items with no embedding were : *agent de police, droit de grâce, grâce presidentielle, mine de crayon, mine antichar, mine antipersonnel, descente en rappel, empourprer, cramoisir, vagir.*

2. http ://espeak.sourceforge.net

Features	Model	Spearman's correlation (ρ)	Pair-based	Rank-based (exact)	Rank-based (± 1)
All features	no HL	**0.616**	0.786	0.490	**0.929**
Characters	no HL	0.307	0.734	0.375	0.680
Frequency	no HL	0.493	**0.812**	**0.606**	0.908
Phonemes	no HL	0.304	0.597	0.231	0.480
Embeddings	no HL	0.516	0.747	0.462	0.857
Freq + emb	no HL	**0.604**	0.773	0.471	0.908
Freq + char	1 HL	0.513	0.805	0.577	**0.929**
Freq + phon	1 HL	0.520	**0.812**	**0.625**	0.908
François et al. (2016)	-	-	0.789	0.596	0.902
Baseline	-	-	**0.820**	0.595	**0.935**

TABLE 2 – Results of the best systems trained on Manulex for each feature combination. Pair- and rank-based accuracy is measured on the manually ranked set of synonyms used for evaluation. Correlation is calculated against the ranking of these words in Manulex. The two best scores for each measure are marked in bold.

French subtitles of 4 types of films and series differing in their original language. Word frequency has been shown to be a very effective predictor of complexity in past work.

— 300-dimensional **word embeddings** [3] previously trained on Wikipedia with fastText using the Skip-Gram model (Bojanowski *et al.*, 2016). Word embeddings encode distributional information of words, and we expect them to be able to capture differences in complexity. Our hypothesis is that complex words appear in more complex texts.

5 Evaluation

We evaluate our systems on the manually ordered synonym lists described in Section 3 measuring the **pair-based** and **rank-based** accuracy of the predictions against human judgments. For the pair-based evaluation, we exhaustively create complex $\rightarrow$ simple pairs of synonyms from every set of synonyms. For example, from the group *forger* $\rightarrow$ *formuler* $\rightarrow$ *former* $\rightarrow$ *inventer*, ordered from complex to simple, we derive 6 pairs : (*forger* $\rightarrow$ *formuler, forger* $\rightarrow$ *former, forger* $\rightarrow$ *inventer, formuler* $\rightarrow$ *former, formuler* $\rightarrow$ *inventer, former* $\rightarrow$ *inventer*), preserving the directionality indicated by the annotators. We expect a good system to assign higher scores to more complex words, producing the correct ordering. We report the proportion of pairs that were correctly ordered by the system.

In the **rank-based** evaluation, the system has to order a full group of synonyms based on the score that it assigns to each word. We report the percentage of cases in which the system correctly predicts the exact position of a synonym. Following François *et al.* (2016), we additionally report the proportion of cases where a synonym was placed in the original place or only one position away. For calculating the latter, we exclude groups of synonyms with only two words, which would otherwise always count as correct (there is only one such pair in the development set and one in the test set). In the case of ties between the scores of two synonyms, they are considered to be incorrectly placed, since this means the system is not able to detect the difference in complexity between the two words.

We test the three proposed architectures (with 1 or 2 HL(s), and without HL) on the development sets extracted from Manulex and FLElex and from the manually-ranked synonym resource. We retain as

3. Available for 294 languages at https ://github.com/facebookresearch/fastText/blob/master/pretrained-vectors.md

Features	Model	Spearman's correlation (ρ)	Pair-based	Rank-based (exact)	Rank-based (± 1)
All features	1 HL	0.429	0.701	0.519	0.816
Characters	1 HL	0.291	0.734	0.375	0.694
Frequency	no HL	**0.569**	0.812	**0.606**	0.908
Phonemes	no HL	0.275	0.597	0.231	0.500
Embeddings	no HL	0.499	0.753	0.558	0.837
Freq + emb	2 HL	0.541	0.688	0.471	0.786
Freq + char	1 HL	**0.596**	0.805	**0.606**	**0.939**
Freq + phon	1 HL	**0.596**	**0.825**	**0.615**	0.908
François et al. (2016)	-	-	0.789	0.596	0.902
Baseline	-	-	**0.820**	0.595	**0.935**

TABLE 3 – Results of the best systems trained on FLElex for each feature combination. Pair- and rank-based accuracy is measured on the manually ranked set of synonyms used for evaluation. Correlation is calculated against the ranking of these words in FLElex. The two best scores for each measure are marked in bold.

Freq + char (correct)	dépouiller → dérober → piquer → voler
Embeddings	dérober → dépouiller → piquer → voler
Freq + char	mental → spirituel → fin
Embeddings (correct)	spirituel → mental → fin

TABLE 4 – Examples of synonyms that were correctly and incorrectly ranked by one of the best performing systems (Freq + char) and the embeddings-based system, both trained on FLElex.

the best model for each feature combination the one that obtains the strongest correlation (Spearman's ρ) with Manulex and FLElex original classes and the highest accuracy scores in the pair-based and rank-based evaluations on the synonym development set. In the case of two models performing equally or very similarly, we choose the simplest one (i.e. the one with fewer layers).

We compare the results of the best model for different feature combinations to the ones produced by François *et al.*'s model on the same test set. The size of the dataset used in our evaluation is slightly different from the one used in their original work, since lexical items that had no embeddings were left out. We recompute the accuracy of their system output on the reduced test set. As a baseline, we use the simple frequency feature with no training involved, which considers more frequent words to be simpler and orders them accordingly.

6 Results and discussion

Results are presented in Tables 2 (Manulex) and 3 (FLElex). The patterns observed and the scores obtained by the systems trained on Manulex and FLElex are, in general, similar. One could expect FLElex to result in a system producing better rankings because of its finer granularity with respect to Manulex, but the much bigger size of the latter probably counteracts this effect.

Among all feature combinations, frequency combined with number of characters or with phonemes seem to be the best predictors of complexity for both Manulex and FLElex. The number of phonemes is the worst performing feature when used on its own, and obtains the lowest score in all measures. This is somewhat surprising, since Gala *et al.* (2014) show that phonemes are slightly more strongly correlated to Manulex and FLElex classes than characters, which do a bit better in this setting but

still are the next poorest performing feature. However, these features alone produce a considerable amount of ties which substantially lower their performance.

The combination of word embeddings with a strong feature such as frequency does not seem to improve the results of embeddings alone on FLElex. The system that relies only on embeddings obtains better results on this dataset than the system using all features. On Manulex, on the contrary, embeddings seem to benefit from being combined with other features. In Table 4 we give an example of a group of synonyms that is correctly ranked by one of the best systems – the one that combines frequency and length in characters (Freq+char) – in the FLElex evaluation, and which the embeddings-based system ranked incorrectly. The lower part of the table shows a case where the opposite happened. We observe that the systems switch words that are situated next to each other in the rankings and which would be more difficult to rank than words situated in the two extremes, given that they might be similarly complex or simple. The difficulty of the task is highlighted by the low inter-annotator agreement reported by François *et al.* (2016) which must be partly due to this type of hard-to-rank cases.

The frequency-based baseline is very powerful and is only beaten with a small margin in the rank-based accuracy evaluation by more complex models that have frequency among their features. François *et al.* (2016)'s system is one of the best performing models, but whilst it employs 21 different features, its accuracy is slightly below that of simpler frequency-based models. This adds more evidence to the established idea that frequency is a crucial indicator of lexical complexity.

7 Conclusions

In this work, we have explored the use of word embeddings alone and combined with other features for lexical complexity prediction in French. Our system learns a complexity score from complexity levels based on age of acquisition (Manulex) and second language proficiency (FLElex). The evaluation on a synonym ordering task seems to indicate that whereas word embeddings obtain better results than other features in isolation, frequency-based systems – even a simple frequency baseline – are better suited for this task. The best performing systems use frequency or combine it with characters or phonemes, two features that have been proven to be useful in past work. In the future, we plan to experiment and compare with other kinds of word embeddings built from corpora with a higher level of stylistic variation.

8 Acknowledgements

We would like to thank the anonymous reviewers for their thoughtful and constructive comments. We particularly would like to thank Núria Gala and Thomas François for the interesting discussions on French lexical complexity resources, and for sharing with us the manually annotated synonyms dataset used for evaluation. This work has been supported by the French National Research Agency under project ANR-16-CE33-0013.

Références

ARYA D. J., HIEBERT E. H. & PEARSON P. D. (2011). The effects of syntactic and lexical complexity on the comprehension of elementary science texts. *International Electronic Journal of Elementary Education*, **4**(1), 107.

BOJANOWSKI P., GRAVE E., JOULIN A. & MIKOLOV T. (2016). Enriching word vectors with subword information. *arXiv preprint arXiv :1607.04606*.

BROOKE J., TSANG V., JACOB D., SHEIN F. & HIRST G. (2012). Building readability lexicons with unannotated corpora. In *Proceedings of the First Workshop on Predicting and Improving Text Readability for target reader populations*, p. 33–39.

CONSEIL DE L'EUROPE (2001). *Cadre européen commun de référence pour les langues*. Paris : Didier.

DRNDAREVIĆ B. & SAGGION H. (2012). Towards automatic lexical simplification in spanish : an empirical study. In *Proceedings of the First Workshop on Predicting and Improving Text Readability for target reader populations*, p. 8–16.

FRANÇOIS T., BILLAMI M., GALA N. & BERNHARD D. (2016). Bleu, contusion, ecchymose : tri automatique de synonymes en fonction de leur difficulté de lecture et compréhension. In *JEP-TALN-RECITAL 2016*, volume 2, p. 15–28.

FRANÇOIS T., GALA N., WATRIN P. & FAIRON C. (2014). FLELex : a graded lexical resource for french foreign learners. In *LREC*, p. 3766–3773 : Citeseer.

GALA N., FRANÇOIS T., BERNHARD D. & FAIRON C. (2014). A model to predict lexical complexity and to grade words (un modèle pour prédire la complexité lexicale et graduer les mots)[in french]. *Proceedings of TALN 2014 (Volume 1 : Long Papers)*, **1**, 91–102.

JAUHAR S. K. & SPECIA L. (2012). Uow-shef : Simplex–lexical simplicity ranking based on contextual and psycholinguistic features. In *Proceedings of the First Joint Conference on Lexical and Computational Semantics-Volume 1 : Proceedings of the main conference and the shared task, and Volume 2 : Proceedings of the Sixth International Workshop on Semantic Evaluation*, p. 477–481.

KIDWELL P., LEBANON G. & COLLINS-THOMPSON K. (2009). Statistical estimation of word acquisition with application to readability prediction. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing*, volume 2, p. 900–909.

LÉTÉ B., SPRENGER-CHAROLLES L. & COLÉ P. (2004). MANULEX : A grade-level lexical database from french elementary school readers. *Behavior Research Methods, Instruments, & Computers*, **36**(1), 156–166.

NEW B., PALLIER C., FERRAND L. & MATOS R. (2001). Une base de données lexicales du français contemporain sur internet : LEXIQUE^{TM}//a lexical database for contemporary french : LEXIQUE^{TM}. *L'année psychologique*, **101**(3), 447–462.

RELLO L., BAEZA-YATES R., DEMPERE-MARCO L. & SAGGION H. (2013). Frequent words improve readability and short words improve understandability for people with dyslexia. In *IFIP Conference on Human-Computer Interaction*, p. 203–219 : Springer.

SHARDLOW M. (2013). A Comparison of Techniques to Automatically Identify Complex Words. In *51st Annual Meeting of the Association for Computational Linguistics Proceedings of the Student Research Workshop*, p. 103–109, Sofia, Bulgaria.

SPECIA L., JAUHAR S. K. & MIHALCEA R. (2012). Semeval-2012 task 1 : English lexical simplification. In *Proceedings of the First Joint Conference on Lexical and Computational Semantics-Volume 1 : Proceedings of the main conference and the shared task, and Volume 2 : Proceedings of the Sixth International Workshop on Semantic Evaluation*, p. 347–355.

Approche Hybride pour la translitération de l'arabizi algérien : une étude préliminaire

Imane Guellil [1, 2] FaicalAzouaou[1] Fodil Benali [1] ala-eddineHachani[1] HoudaSaadane[3]
(1) Ecole nationale Supérieure d'Informatique, BP 68M, 16309, Oued-Smar, Alger,
(2) Ecole Supérieure des Sciences Appliquées, Alger, Algérie, http://www.essa-alger.dz
(3) GEOLSemantics, 12 Avenue Raspail, 94250 Gentilly, France,
i_guellil@esi.dz, f_azouaou@esi.dz, df_benali@esi.dz, da_hachani@esi.dz, houda.saadane@geolsemantics.com

RESUME

Dans cet article, nous présentons une approche hybride pour la translitération de l'arabizi algérien. Nous avons élaboré un ensemble de règles permettant le passage de l'arabizi vers l'arabe. Á partir de ces règles nous générons un ensemble de candidats pour la translitération de chaque mot en arabizi vers l'arabe, et un parmi ces candidats sera ensuite identifié et extrait comme le meilleur candidat. Cette approche a été expérimentée en utilisant trois corpus de tests. Les résultats obtenus montrent une amélioration du score de précision qui était pour le meilleur des cas de l'ordre de 75,11%. Ces résultats ont aussi permis de vérifier que notre approche est très compétitive par rapport aux travaux traitant de la translitération de l'arabizi en général.

ABSTRACT

A hybrid approach for the transliteration of Algerian Arabizi: A primary study
In this paper, we present a hybrid approach for the transliteration of the Algerian Arabizi. We define a set of rules for the passage from Arabizi to Arabic. Through these rules, we generate a set of candidates for the transliteration of each Arabizi word into arabic. Then, we extract the best candidate. This approach was evaluated by using three test corpora, and the obtained results show an improvement of the precision score which is equal to 75.11% for the best result. These results allow us to verify that our approach is very competitive comparing to others works that treat Arabizi transliteration in general.

MOTS-CLES : arabizi, Dialecte algérien, arabizi algérien, Translitération.
KEYWORDS: Arabizi, Algerian Dialect, Algerian Arabizi, Transliteration.

1 Introduction

L'arabizi est une orthographe spontanée utilisée pour s'exprimer en dialecte arabe combinant lettres latines, chiffres et autres symboles tels que les signes de ponctuations (Al-Badrashiny et al., 2014; Guellil, AZOUAOU, 2017) . L'arabizi est généralement utilisé par les locuteurs arabes pour les échanges dans les réseaux sociaux, les chats voir les SMS. Néanmoins, comme la plupart des outils de traitement des dialectes arabes, et du dialecte algérien (DALG) en particulier, sont dédiés aux messages écrits en lettres arabes, par conséquent l'arabizi ne peut donc être traité comme tel. La plupart des recherches s'oriente vers la transformation de l'arabizi vers l'arabe. Cette transformation

est nommée *la translitération*. La translitération est un processus de passage d'un texte écrit en un script ou alphabet donné vers un autre (Guellil et al., 2017a; Guellil et al., 2017b; Josan, Lehal, 2010; Kaur, Singh, 2014). La translitération de l'arabizi vers l'arabe fait cependant face à un ensemble de problématiques :

1) Traitement des voyelles : les voyelles (a, i, o, u, e, y) peuvent être remplacées par les différentes lettres arabes (أ,ا,ي,و,ة) ou encore par aucune lettre. Cela dépend de leurs emplacements dans le mot.

2) L'ambiguïté entre plusieurs lettres : une lettre arabizi peut correspondre à plusieurs lettres arabes. Par exemple, la lettre 't' peut correspondre aux deux lettres arabes 'ت' t et 'ط' T.
3) L'ambiguïté reliée au contexte : dans certains cas, plusieurs translitérations peuvent correspondre au même mot. Par exemple le mot 'matar' pourrait être translitéré en 'مطر' *maTar*[1] 'la pluie' ou encore 'مطار' *maTaAr* 'aéroport' (Al-Badrashiny et al., 2014).
4) **Ambiguïté reliée au code switching[2] :** certains mots d'autre langues tels que le français ou l'anglais peuvent être pris pour des mots en arabizi. Par exemple le mot 'men' en anglais.

Afin d'adresser ces problématiques, nous présentons dans cet article une approche hybride pour la translitération de l'arabizi algérien. Dans la suite du présent article, nous présentons dans la section 2 une synthèse de l'état de l'art liée à la problématique de la translittération, puis nous mettons en avant dans la section 3 l'originalité de notre approche par rapport aux travaux étudiés. Ensuite nous décrivons dans la section 4 notre approche proposée. Quant à la section 5, elle sera consacrée à la présentation des expérimentations menées ainsi que les résultats obtenus. Enfin, dans la section 6 nous concluons cet article avec une présentation de nos travaux futurs.

2 État de l'art

Le problème de la translittération a suscité l'intérêt des spécialistes dans plusieurs langues. Cet intérêt récent est lié au développement croissant de l'utilisation d'Internet (Saâdane, Semmar, 2012) Trois approches sont couramment évoquées dans la littérature pour réaliser la translittération: 1) A base de règles. 2) Statistiques et 3) Hybrides combinant les deux précédentes (Guellil et al., 2017; Kaur, Singh, 2014). Habash, et al., (2007), ont proposé un ensemble de règles permettant le passage de l'arabizi vers l'arabe. Ils ont signalé un nombre d'exceptions et de défis reliés aux traitements des voyelles. Rosca&Breuel (2016) ont abordé l'approche statistique où ils ont présenté un modèle basé sur les réseaux de neurones pour effectuer la translitération entre plusieurs paires de langues dont l'arabe et l'anglais. L'approche hybride est utilisée dans les travaux de Al-Badrashiny et al., 2014; Darwish 2013; Guellil et al., 2017a; Guellil et al., 2017b; Mayet al., 2014; Saâdane et al., 2017; Saâdane et al., 2013; van der Wees et al., 2016). Tous ces travaux suivent la même idée générale, à savoir générer un ensemble de possibilités de translittération, appelés candidats, pour ensuite déterminer le meilleur candidat à l'aide d'un modèle de translitération ou autre. Pour ce faire, Darwish (2013) construit manuellement un ensemble contenant 3452 mots arabizi (extrait de Twitter) translitéré vers l'arabe. Une partie de ce corpus arabizi-arabe a été utilisée dans le travail de (Al-Badrashiny et al., 2014). Les auteurs de ce travail font cependant appel à un automate d'état fini pour le passage de l'arabizi vers l'arabe. Dans les travaux de (May et al., 2014; van der Wees et al., 2016), les auteurs analysent également les résultats de la traduction après la phase de translitération. Enfin, les travaux de (Guellil et al., 2017; Saâdane et al., 2017; Saâdaneet al., 2013) constituent les

[1]Translittération arabe présentée dans schemeHabash-Soudi-Buckwalter (HSB) (Habash et al., 2007)

[2] Code switching: Présence de plusieurs langues ou dialectes au sein du même message

uniques références que nous avons recensées sur l'arabizi algérien. Guellil et al., (2017) ont développé un algorithme basé sur les règles pour la translitération en s'appuyant sur un corpus arabe qu'ils ont translitéré en arabizi. Ces auteurs ajoutent ensuite la notion de traduction automatique de l'arabizi après la translitération (Guellil et al., 2017). Saâdane et al., (2017); Saâdane et al., (2013) ont translitéré le texte arabizi en arabe en utilisant un automate d'état fini. Les résultats générés sont normalisés en suivant la convention de transcription nommée *CODA* "Conventional Orthography for Dialectal Arabic" (Saâdane, Habash, 2015). Ensuite les nouveaux résultats sont filtrés en utilisant un analyseur morphologique de l'arabe. Nous signalons cependant que ces travaux se focalisent plus sur l'identification de l'origine dialectale.

3 Positionnement de notre approche par rapport aux travaux étudiés

Nous présentons dans cet article une approche de translitération de l'Arabizi Algérien vers l'arabe. Pour définir cette approche, nous nous sommes appuyés sur plusieurs travaux comme suit :

1) Nous avons utilisé une table de passage de l'arabizi vers l'arabe similaire à celle présentée dans les travaux de van der Wees et al., (2016), qui est extraite de Wikipédia[3]. Cependant, nous avons défini en plus des règles de passage ainsi que nos propres règles dédiées au traitement de l'Arabizi Algérien, notamment les différentes ambiguïtés de translitération. Pour ce faire, nous nous sommes inspirés des travaux de (Guellil et al., 2017) décrivant les caractéristiques de l'Arabizi Algérien.

2) Les travaux de Al-Badrashiny et al., (2014); Darwish (2013); Guellilet al., (2017a); Guellil, et al., (2017b); May et al., (2014); van der Wees et al.,(2016) génèrent un ensemble de candidats pour la translitération d'un mot arabizi en arabe. Par exemple, les candidats générés des travaux de ces auteurs pour le mot '3afsa' (une astuce) sont : 'عافسة' *ɛaAfsaħ* , ' عافسا' *ɛaAfsaA*, 'عافزة' *ɛaAfzaħ*, 'عافزا'*ɛaAfzaA*, etc. Néanmoins nous ne trouvons aucun candidat sous la forme de 'عفسة' qui est la translitération correcte de ce mot. La valeur ajoutée de notre travail est que notre algorithme est capable de générer de tels candidats en remplaçant les voyelles par le caractère vide (NULL).

3) Les travaux de (Darwish 2013; Guellil et al., 2017a; Guellilet al., 2017b) qui font appel à un modèle de translitération pour déterminer le meilleur candidat. Cependant, ces approches dépendent d'un corpus parallèle, correspondant à la translitération d'un ensemble de messages de l'arabizi vers l'arabe, en plus du coût élevé de la réalisation de ces même corpus. Pour notre part, au lieu de construire des corpus parallèles arabizi-arabe, nous appliquons un modèle de translitération sur un corpus Arabe assez volumineux extrait des médias sociaux (par nos soin) et comparant ainsi les résultats obtenus.

4 Approche de translitération de l'arabizi algérien vers l'arabe

Notre approche se compose de quatre phases importantes : 1) Extraction et prétraitement du corpus Arabe et des messages. 2) Proposition et application des règles pour l'arabizi algérien. 3) Génération des différents candidats et 4) Extraction du meilleur candidat. Nous présentons dans la Figure 1 l'architecture générale de notre approche.

[3]https://en.wikipedia.org/wiki/Arabic_chat_alphabet

4.1 Extraction et prétraitement du corpus Arabe et du message arabizi

Nous commençons tout d'abord par l'extraction d'un corpus Arabe issu des réseaux sociaux et contenant un ensemble de commentaires de locuteurs Algériens. Pour ce faire, nous avons ciblé un ensemble de pages très populaires en Algérie comme la page **Ooredoo**[4] (opérateur téléphonique). Après l'extraction de ce corpus, nous nous sommes focalisés sur les messages écrits uniquement en caractères arabes. Nous supprimons ensuite l'exagération (par exemple, le mot hiaaaaati–*ma vie*-est transformé en hiati) de ce corpus et nous remplaçons les différents caractères arabes par leurs Unicodes. Nous procédons ensuite aux mêmes prétraitements sur le message arabizi

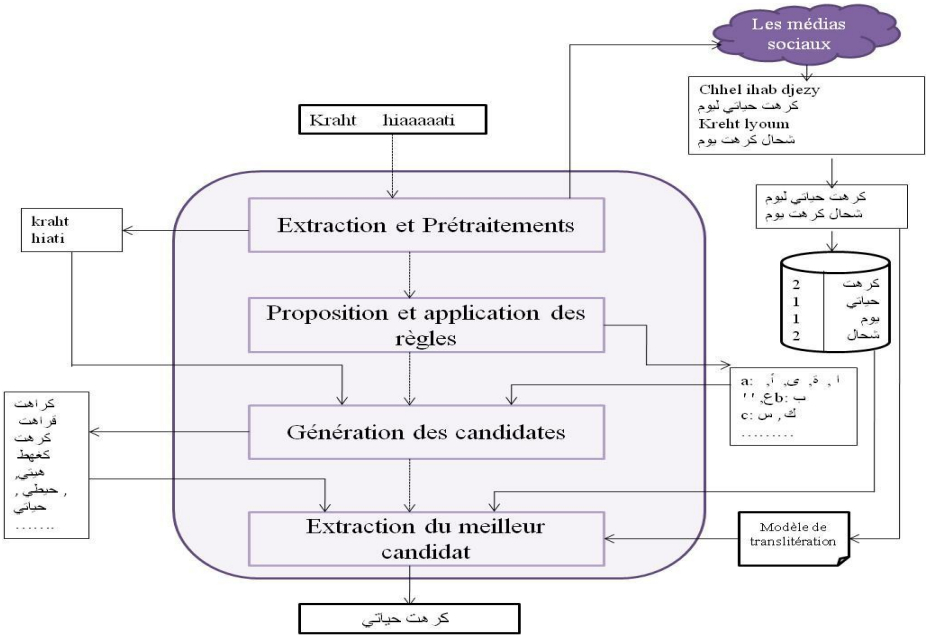


Figure *1*: Architecture générale de notre approche

4.2 Proposition et application des règles pour l'arabizi algérien

Nous présentons dans le tableau 1, les différentes possibilités de remplacement de chaque lettre en arabizi algérien. En plus de ce tableau, nous définissons un ensemble de règles de passage de l'arabizi vers l'arabe. Par exemple : la voyelle 'a' est remplacée par la lettre 'أ' au début. Au milieu du mot, cette lettre peut être remplacée soit par le caractère Arabe 'ا' ou encore par le caractère vide.

4.3 Génération des différents candidats

En appliquant les différents remplacements du tableau 1 ainsi que des différentes règles élaborées, chaque mot arabizi donne naissance à plusieurs mots en arabe. Pour illustrer cela, reprenons l'exemple des deux mots « kraht » et « hiati ». Le mot « kraht » donne lieu à 32 candidats. Parmi

[4]https://fr-fr.facebook.com/OoredooDZ/

ces derniers, nous citons: كراهت *krAht*, قراهت *qrAht* ,كرهت *krht*, كغهط *kγhT*,قرحت *qrHt*, etc. Le mot correctement translitéré étant «كرهت» *krht*. Quant au mot « hiati », nous avons 16 candidats parmi lesquels nous citons :هياتي *hyAty*, هيتي *hyty*, حيطي *HyTy* , حياتي *HyAty*, حيتي *Hyty*, etc. Le mot correctement translitéré étant «حياتي *HyAty*».

Lettre en Arabizi	Lettre en arabe	Lettre en Arabizi	Lettre en Arabe	Lettre en Arabizi	Lettre en Arabe
A	ا, ة, ى, أ, ع "	k	ك, ق	U	و, أ"
B	ب	l	ل	V	ف
C	س , ك	m	م	W	و
D	د, ض, ظ	n	ن	X	كس
E	ا"	o	و , "أ	Y	إ, ي"
F	ف	p	ب	Z	ز
G	ق	q	ك	7	ح
H	ه , ح	r	ر,غ	5	خ
I	ي"	s	س , ص	3	ع
J	ج	t	ت ط	9	ق

Table 1 : Lettres de passage de l'arabizi vers l'Arabe

4.4 Extraction du meilleur candidat à la translitération

Pour extraire le meilleur candidat de translitération d'un mot arabizi, nous réalisons les deux étapes suivantes : 1) réalisé une recherche simple de chaque candidat au sein de notre corpus arabe, et 2) effectuer une recherche basée sur un modèle de langue appliqué sur notre corpus arabe. Au cours de la recherche simple, nous recherchons chaque candidat au sein de notre ensemble de mots afin de récupérer le nombre d'occurrence de chaque candidat. Pour la recherche basée sur un modèle de translitération, nous recherchons chaque candidat au sein de notre modèle en extrayant la probabilité de chacune. Nous retournons ensuite le candidat ayant la probabilité la plus élevée.

5 Expérimentations et résultats

5.1 Le corpus arabe utilisé

Pour la création de nos ressources linguistiques (corpus), nous avons tout d'abord ciblé 133 pages Facebook Algérienne dont : *Ooreedo, Djezzy, HamoudBoualem,* etc. Ces pages concernent les opérateurs téléphoniques, des producteurs de limonade ou de jus de fruit, la presse, etc.

Nous avons également découpé le corpus PADIC (Meftouh, Harrat, Jamoussi, Abbas, & Smaili, 2015) en un ensemble de termes et construit un dictionnaire du dialecte algérien, que nous avons pu récolter grâce à la traduction d'un dictionnaire anglais en faisant appel à l'API glosbe[5]. A l'aide de l'API de Facebook (RestFB[6]), et grâce à la fonction « *search»* de RestFB, nous extrayons l'ensemble des posts et commentaires relatifs aux pages et différents dictionnaires. Par ailleurs, nous signalons que les sources censées alimenter notre corpus concernent divers sujets : politique, sport,

[5] https://glosbe.com/

[6] http://restfb.com/

économie, etc. Notre corpus contient **3 668 575** messages et ce après prétraitement et extraction des messages écrits en caractère arabe.

5.2 Les corpus de test utilisés

Nous avons fait appel à trois corpus de tests :
1) Test_300: Contenant 300 messages (2005 mots) extrait de Facebook. Ces messages font partie du corpus contenant **3 668 575** que nous avons extrait précédemment.
2) Test_50 : Contenant 50 messages (527 mots) messages extrait du corpus de Cotterell, et al.,(2014). Ce corpus étant le seul corpus de l'arabizi algérien en libre accès au sein de la communauté de recherche. Ce corpus a déjà été utilisé comme corpus de test au sein des travaux dans (Guellil et al., 2017).
3) Test_200 : Contenant 200 messages (933 mots) extraient du corpus Parallèle PADIC (Meftouh et al., 2015). Néanmoins ce corpus est en caractère arabe mais il a été translitéré vers l'arabizi dans les travaux de Guellil et al., (2017).

5.3 Les résultats expérimentaux

Pour nos expérimentations, nous découpons notre corpus arabe en plusieurs parties. Chaque découpage a été utilisé pour entraîner un modèle de translitération et pour faire la recherche simple basée sur notre algorithme. Nous menons ainsi plusieurs expérimentations où nous utilisons respectivement : 1% (36 682 messages/ 63 269 mots), 5%(183 413 messages/ 177 722 mots), 10% (366 827 messages/ 268 751 mots), 25% (917 068 messages/ 454 817 mots), 50% (1 834 137 messages/ 658 738 mots), 75% (2 751 205 messages/ 810 611 mots) et finalement 100% (3 668 275 messages/ 930 462 mots) de notre corpus arabe. Pour le modèle de translitération nous faisons appel à l'implémentation JAVA du modèle KenLM[7]. Nous testons notre algorithme pour différents n-gramme (avec n allant de 2 à 5). Nous nous sommes rendus compte que nous obtenions pratiquement les mêmes résultats à chaque fois, car la recherche du meilleur candidat se fait en 1-gramme uniquement. De ce fait, nous avons décidé d'utiliser un modèle de translitération avec 2-gramme. Nous présentons dans le tableau 2, l'ensemble des résultats de translitération de l'arabizi algérien en se basant sur les deux approches décrites précédemment (recherche simple et modèle de translitération avec kenLM). Nous calculons pour chaque corpus de test l'accuracy qui est défini comme suit :
l'Acurracy= nombre de mot correctement translitéré/ nombre total de mot dans le corpus

Approche	Corpus	1%	5%	10%	25%	50%	75%	100%
Recherche simple	Test_50	67.36	70.02	72.49	73.24	74.19	74.57	**74.76**
	Test_200	67.31	69.74	69.67	70.95	71.06	71.28	72.03
	Test_300	68.79	72.02	72.97	73.72	74.16	74.71	**75.11**
KenLM	Test_50	66.79	70.21	71.54	72.87	72.49	72.30	72.67
	Test_200	64.84	66.88	67.85	68.38	69.45	69.24	69.34
	Test_300	68.33	70.57	71.73	72.37	72.76	73.07	73.27

Table 2 : Résultats de translitération de l'Arabizi algérien

[7] https://github.com/jbaiter/kenlm-java

5.4 Analyse des résultats et des cas d'erreurs

D'après le tableau 2, nous constatons que la taille du corpus arabe influe sur les résultats obtenus. Plus ce corpus est volumineux, meilleurs sont les résultats. Concernant les corpus de test, nous avons utilisé trois corpus contenant respectivement (50, 200 et 300) messages. Nous constatons une amélioration remarquable au sein du corpus Test_50 où la précision dans Guellil et al., (2017)) est à 45.35% et dans notre cas elle atteint 74,76% dans le cas de la recherche simple et où le corpus arabe utilisé est complet (c'est-à-dire 100%). Nous obtenons une précision égale à 75,11% dans le cas de notre corpus Test_300, qui représente le meilleur résultat obtenu, ce qui est compréhensible vu que notre approche est basée sur la translitération des messages extraient des médias sociaux et que la translitération est faite de l'arabizi vers l'arabe et non pas l'inverse comme cela est fait dans dans (Guellil et al., (2017)).

Néanmoins en analysant le corpus translitéré, nous avons identifié les erreurs suivantes : 1) Omission de certaines voyelles où elles devraient apparaître. Par exemple : le mot 'bik' est translitéré en '**بك**' au lieu de '**بيك**'. 2) Présence de certaines voyelles alors qu'elles ne devraient pas apparaître. Par exemple le mot 'bark' est translitéré en '**بارك**' au lieu de '**برك**', le mot 'lawel' est translitéré en '**لاول**' au lieu de '**لول**'. 3) Dans certains cas deux translitérations sont correctes, tout dépend du contexte et du sens de la phrase. Par exemple, le mot 'raht' pourrait être translitéré en '**رحت**' ou en '**راحت**', ou encore le mot 'djabat' qui pourrait être translitéré en '**جبت**' ou en '**جابت**' et ce tout dépend du sens de la phrase. 4) Des erreurs reliées aux mots puisant leurs signification du français et donc non reconnu par notre corpus dans la plupart des cas. Par exemple, le mot 'lafichage' est translitéré en '**لافيشاج**' au lieu de '**لفيشاج**' et le mot 'elsemastar' est translitéré en '**السوماستر**' au lieu de '**السمستر**'.Toutes ces erreurs sont causé par deux principales raisons : 1) La non prise en considération du contexte du mot dans la phrase. 2) Au non traitement des mots ayant comme signification une langue étrangère (principalement le français).

6 Conclusion et perspectives

Dans cet article, nous avons présenté une approche hybride de translitération de l'arabizi algérien vers l'arabe. Cette approche est basée sur la combinaison entre règles et modèles statistiques pour déterminer le meilleur candidat répondant à la translitération d'un mot en arabizi. Notre approche pourrait cependant être améliorée en y intégrant les points suivants :

- L'utilisation d'un corpus plus volumineux pour améliorer les résultats obtenus car nous avons constaté que la taille du corpus influe sur les résultats renvoyés. Il serait également intéressant de se pencher sur le niveau caractère.
- Mis en place d'une approche qui formerait des candidats contenant des n-gramme et non seulement des 1-gramme. Ceci nous aidera à situer le mot au sein de la phrase et non pas le traiter comme entité seule.
- Traiter le cas des mots étranger, par exemple, les mots français.
- Utiliser cette approche pour générer un corpus parallèle arabizi-arabe de manière semi-supervisée. Ce corpus pourrait être utilisé pour générer un modèle statistique.

Enfin, nous signalons que les corpus développés dans le cadre de cette étude seront bientôt mis à la disposition de la communauté scientifique.

Remerciements

Les premiers auteurs sont soutenus par l'Ecole Supérieure des sciences appliquées d'Alger (ESSAA) ainsi que l'Ecole Supérieure d'informatique ESI Alger. Le dernier auteur est soutenu par la DGE (Ministère de l'Industrie) et par la DGE (Ministère de l'économie):projet "DRIRS", référencé par le N :172906108.

Références

Al-Badrashiny M., Eskander R., Habash N., & Rambow, O. (2014). Automatic Transliteration of Romanized Dialectal Arabic. Paper presented at the *CoNLL*.

Cotterell R., Renduchintala A., Saphra N., &Callison-Burch, C. (2014).An algerianarabic-french code-switched corpus. Paper presented at the *Workshop on Free/Open-Source Arabic Corpora and Corpora Processing Tools Workshop Programme*.

Darwish K. (2013). Arabizi detection and conversion to Arabic.*arXiv preprint arXiv:1306.6755*.

Guellil I., & Azouaou F. (2017). ASDA: Analyseur Syntaxique du Dialecte Algérien dans un but d'analyse sémantique. *arXivpreprint arXiv:1707.08998*.

Guellil I., Azouaou F., & Abbas M. (2017). Comparison between Neural and Sta-tistical translation after translitera-tion of Algerian Arabic Dialect. Paper presented at the *WiNLP: Women & Underrepresented Minorities in Natural Language Processing* (co-located withACL 2017).

Guellil I., Azouaou F., Abbas M., & Fatiha, S. (2017a). Arabizi transliteration of Algerian Arabic dialect into Modern Standard Arabic. Paper presented at the *Social MT 2017/First workshop on Social Media and User Generated Content Machine Translation*.

Guellil I., AZOUAOU F. (2017b). Neural Vs Statistical Translation of Algerian Arabic Dialect written with Arabizi and Arabic letter. The 31st *Pacific Asia Conference on Language, Information and Computation PACLIC.*

Guellil., I et AZOUAOU F. (2017). Bilingual Lexicon for Algerian Arabic Dialect Treatment in Social Media. *WiNLP: Women & Underrepresented Minorities in Natural Language Processing (co-located with ACL 2017)*

Habash N., Soudi A., &Buckwalter, T. (2007). On arabic transliteration *Arabic computational morphology* (pp. 15-22): Springer.

Josan G. S., &Lehal, G. S. (2010).A Punjabi to Hindi Machine Transliteration System.*Computational Linguistics and Chinese Language Processing, 15*(2), 77-102.

Kaur K., & Singh P. (2014).Review of Machine Transliteration Techniques.*International Journal of Computer Applications, 107*(20).

May J., Benjira Y., &Echihabi, A. (2014). An Arabizi-English social media statistical machine translation system. Paper presented at *the Proceedings of the 11th Conference of the Association for Machine Translation in the Americas*.

Meftouh K., Harrat S., Jamoussi S., Abbas, M., &Smaili, K. (2015). Machine translation experiments on PADIC: A parallel Arabic dialect corpus. Paper presented at the *The 29th Pacific Asia conference on language, information and computation*.

Saâdane H., &Habash, N. (2015).A conventional orthography for Algerian Arabic.In *Proceedings of the Second Workshop on Arabic Natural Language Processing* (pp. 69-79).

Saâdane H., & Semmar N. (2012).Utilisation de la transliteration arabe pour l'amélioration de l'alignement de mots à partir de corpus parallèles français-arabe (UsingArabic Transliteration to Improve Word Alignment from French-ArabicParallelCorpora)[in French].Paper presented at *the Proceedings of the Joint Conference JEP-TALN-RECITAL 2012, volume 2: TALN*.

Saâdane H., Nouvel,. D., Seffih H., &Fluhr, C. (2017). Une approche linguistique pour la détection des dialectes arabes. Paper presented at the 24e *Conférence sur le Traitement Automatique des Langues Naturelles (TALN)*.

Rosca M., &Breuel T. (2016). Sequence-to-sequence neural network models for transliteration. *arXiv preprint arXiv:1610.09565*.

Saâdane H., Guidere M., &Fluhr C. (2013).La reconnaissance automatique des dialectes arabes à l'écrit. Paper presented at the *Colloque International Traduction et Champs Connexes, Quelle Place Pour La Langue Arabe Aujourd'hui.*

van der Wees, M., Bisazza A., &Monz C. (2016). A Simple but Effective Approach to Improve Arabizi-to-English Statistical Machine Translation. *WNUT 2016*, 43.

Lieu et nom de lieu, du texte vers sa représentation cartographique

Catherine Dominguès
Univ. Paris-Est, LASTIG COGIT, IGN, ENSG, 73 avenue de Paris, F-94160 Saint-Mandé
catherine.domingues@ign.fr

RESUME

Les lieux constituent une information structurante de nombreux textes (récits, romans, articles journalistiques, guides touristiques, itinéraires de randonnées, etc.) et leur recensement et leur analyse doit tenir compte des aspects thématiques abordés dans les textes. Le travail proposé ici s'inscrit dans les domaines de la linguistique de corpus et de la cartographie. La définition de lieu est augmentée de celle d'objet localisé et la désignation de ces lieux peut alors être construite sur un nom propre ou un nom commun. Des expérimentations sont menées afin d'identifier les lieux noms propres avec des gazetiers et les lieux noms communs grâce à un modèle d'apprentissage automatique. Les résultats sont discutés sous la forme d'une comparaison entre les caractéristiques linguistiques des noms de lieux et les propriétés visuelles que devront satisfaire leur représentation cartographique.

ABSTRACT

Place and place name, from text to its cartographic portrayal.
Places constitute structuring information for numerous texts (narratives, novels, journalistic articles, tourist guides, trip itineraries, etc.) and the inventory and analysis of the places must take into account the thematic features addressed in the texts. This work is in the fields of corpus linguistics and cartography. The definition of place is broadened to located object which is referred to by a common noun. The designation of this expanded definition place may be made from proper nouns or commons nouns. Experiments which are described enable to identify the proper noun place names by means of gazetteers and the common nouns by means of machine learning models. The results are discussed in the form of comparison between the linguistic features of place names and the visual properties which their cartographic layout will have to fulfil.

MOTS-CLES : lieu, nom de lieu, nom propre, nom commun, linguistique de corpus, cartographie.
KEYWORDS: place, place name, proper name, common noun, corpus linguistics, cartography

1 Introduction

Les lieux constituent une information structurante (Caquard, Fiset, 2013) pour de nombreux textes (récits, romans, articles journalistiques, guides touristiques, itinéraires de randonnées, etc.). Ils sont« *des éléments informationnels pertinents dont on parle et qui jouent un rôle dans la*

description d'un événement, d'un fait » (Nouvel *et al.*, 2015:13). Pour certains textes, les lieux et des objets localisés, pertinents dans le contexte, ainsi que les propriétés (géographiques, sociologiques, émotionnelles, etc.) qui leur sont associées constituent un mode d'exploration prépondérant. Le travail présenté ici s'appuie sur des textes appartenant à cette catégorie : des récits de vie de Républicains espagnols et des commentaires concernant des projets d'aménagement urbain. Bien que d'apparences très dissemblables, ces textes partagent une même définition élargie du lieu (Brando *et al.*, 2016), espace géographique ou objet localisé, et donc présentent les mêmes particularités pour leur identification : noms propres (désormais NPr) comme *Espagne* ou *Paris*, mais aussi syntagmes nominaux contenant un NPr (Gaio *et al.*, 2012) et désignant, dans le contexte du texte, un lieu unique et localisable comme *gave de Pau*, et enfin syntagme nominal sans nom propre[1] et dont la localisation n'est pas toujours définie ou accessible : *usine de textile*, *zone de commerce*. L'analyse de ces textes vise à aider à leur représentation cartographique i.e. la géo-localisation des lieux évoqués dans les textes et la symbolisation de leurs propriétés, en particulier émotionnelles[2]. Cette représentation doit en outre tenir compte de la granularité des lieux mentionnés, variable selon les textes ou les extraits des textes : *Bordeaux*, en tant que ville, est évoqué de nombreuses fois dans les récits des Républicains espagnols, aussi bien positivement que négativement, mais des lieux de la ville sont aussi mentionnés : la *rue Naujac*, refuge chez le grand-père d'un locuteur, et donc connoté positivement ; le *fort du Hâ*, lieu de torture, et le *siège de la Kommandantur* associés à des perceptions négatives.

La définition d'un lieu et de sa désignation sont donc à replacer dans un domaine nécessairement transdisciplinaire : traitement automatique des langues (TAL), géographie, histoire, sciences politiques, géomatique, sociologie, et la recherche d'information concernant les lieux doit tenir compte de ces différentes facettes pour être pertinente et efficace.

Le travail présenté ici s'inscrit dans le domaine de la linguistique de corpus et s'intéresse à la désignation d'un lieu et l'identification de ses propriétés caractéristiques en contexte, avec pour objectif de le cartographier de manière pertinente. Des définitions de lieu sont d'abord rappelées, ce qui conduit à construire la désignation d'un lieu à l'aide de nom propres et de noms communs, le plus souvent génériques. La partie 3 décrit les expérimentations qui ont été menées afin de vérifier ces propositions. Ces propositions sont discutées dans la partie 4 qui se conclut par des perspectives.

2 Lieu et nom de lieu

Le lieu n'est pas une donnée géographique mais est construit socialement ; sa désignation relève du NPr et du nom commun (désormais Nc), et doit guider sa représentation cartographique.

---

1 La campagne QUAERO s'est aussi intéressée aux entités nommées dont les noms sont construits exclusivement sur des noms communs par exemple *13ème arrondissement* mais, dans un contexte de ville donné, *13ème arrondissement* désigne un lieu unique et identifiable. Il n'en est pas de même pour la mention *usine de textile* qui, sans précision discriminante supplémentaire, peut désigner différents lieux dans la zone décrite par le texte.

2 Différents projets visant cet objectif ont été proposés : *Pour une cartographie émotionnelle des récits de vie*, porté par Sébastien Caquard (université de Concordia, Montréal), subventionné par le Fonds de recherche société et culture du Québec en 2015 ; *MATRICIEL - Lieux des migrants à travers des récits de vie : mots, perceptions, émotions, cartes*, projet exploratoire premier soutien (PEPS), subventionné par l'université Paris-Est et le CNRS que nous avons porté en 2016.

La définition d'un lieu prend place dans le domaine de la géographie. Des géographes, par exemple Brunet *et al.* (1993:485), et des linguistes, par exemple Dubois *et al.* (1994:485), s'accordent sur le fait que « *la matière est généralement divisée selon la géographie* » en montagnes, fleuves et étendues d'eau, lieux habités, etc. Mais l'expérience quotidienne montre que tous les lieux ne sont pas nommés. Van de Velde (2000:38) relève deux activités qui contribuent à la création d'un lieu : l'habitat et la mobilité. Construire un lieu constitue un processus ancré socialement qui permet de distinguer, d'une manière qui vise à être partagée et pérenne, une portion de territoire et de la repérer sur la terre (Brunet *et al.*, 1993:485).

Un lieu est désigné par son nom mais tous les noms ne dénotent pas des lieux. Ces noms sont liés à un référentiel terrestre et permettent de repérer les lieux dans ce référentiel. Pour Van de Velde (2000:38) ces noms doivent « *dénoter des choses ayant un rapport essentiel avec la terre* ». Les noms qui désignent des lieux peuvent, selon les cas, entrer dans les catégories des NPr ou des Nc[3].

Par ses définitions, ses propriétés et ses emplois, le NPr est bien adapté à la désignation des lieux. Il permet de distinguer et désigner, par une forme définie dans la langue et de manière conventionnelle et stable, un élément qui émerge de son environnement par des considérations géographiques et/ou sociales. Il est un désignateur rigide (Kripke, cité par (Recanati, 1983)) car ce lien de désignation perdure en dehors de toute situation de communication (Kleiber, 1996:573). Il constitue un NPr pur (Jonasson, 1994), (Calabrese-Stienberg, 2009:4). Il est opaque (Jonasson, 1994:36) car il ne décrit pas les propriétés de l'objet qu'il désigne (Kleiber (1996:573) : par exemple, les noms *Palais de Tokyo* (musée situé à Paris) ou *Espagne* renseignent peu sur les propriétés, actuelles ou celles actives dans le contexte de la désignation, des objets désignés.

Bien que le NPr constitue un « lien stable et direct avec un objet » (Calabrese-Stienberg, 2009:7), il est possible d'observer un changement du référé sans changement du référent. Dans le cas d'un pays, celui-ci peut enregistrer des annexions ou des pertes sans changer de nom ; à l'inverse, le référent NPr peut changer sans que le lieu référé ne change, du moins dans ses frontières. Cependant, le plus souvent, les changements de nom traduisent des changements politiques ou administratifs de l'entité considérée : par exemple, le Honduras britannique qui devient le Belize au moment de son indépendance. Ces changements de désignation procèdent de la « *renégociation du contenu symbolique du toponyme* ». Le NPr est monoréférentiel[4] mais il acquiert une dimension polyréférentielle en discours (Auboussier, 2016). Paveau (2008) évoque la « *souplesse du toponyme* » dans le réseau cognitif de chacun. « *Les noms propres constituent les points fixes de l'organisation symbolique, c'est-à-dire en même temps de l'organisation mentale et de la structure du monde* » (Molino, 1982:19) cité par (Paveau, 2008). Cependant, cette polyréférentialité se manifeste différemment selon la catégorie du toponyme. Par exemple pour les noms de pays, la valeur polyréférentielle et symbolique du nom est utilisée comme « *outil de représentation sociale* » (Cislaru, 2008), le nom de pays désignant aussi bien le territoire national que l'état construit sur ce territoire ou le peuple qui l'occupe.

Les éléments remarqués, distingués et désignés par un NPr contribuent à la connaissance et la structuration du monde physique (Jonasson, 1994:18), mais ces NPr ne permettent pas de désigner

[3] Même s'il est impossible de définir des critères de reconnaissance des noms propres toujours fiables (Leroy, 2004 :8-24), cette catégorisation reste opérationnelle pour notre travail.

[4] Dans un contexte donné, le lecteur associera la chaîne *Paris* à la ville française ou togolaise, celle du Kentucky ou du Texas, au lac canadien ou à la montagne colombienne. Pour des outils de TAL, la prise en compte du contexte des entités nommées spatiales et l'extraction des informations pertinentes permettant d'identifier ce contexte restent un problème difficile.

l'ensemble des lieux et objets localisés utiles à l'activité humaine. D'un point de vue linguistique, Van de Velde (2000:38) remarque que tout nom de chose matérielle peut remplir dans un énoncé le rôle sémantique de *lieu*. Blidon (2008:4) rappelle que « *d'une certaine façon, tout objet est géographique si son traitement l'est* » ; un objet qui est géolocalisé, comparé sur des critères de localisation, d'implantation, de classification, à des objets géographiques peut dès lors être considéré comme un lieu. Ce lieu (*zone de commerce*, *axe piétonnier*, *voie cycliste*, *camp de concentration*), cet objet localisé (*banc*, *poubelle*, *chêne centenaire*) sont désignés par une dénomination descriptive, mais qui n'est ni stable ni unique ; cette désignation de lieu ne relève donc plus du NPr mais du Nc.

Cette définition élargie de lieu est aussi dictée par les contextes de production des textes et les objectifs de leur analyse. Ces contextes et objectifs étant spécifiques à chaque situation de production et d'analyse, les éléments qu'il est pertinent d'analyser comme des lieux ou des objets localisés varient selon les situations. Dans le contexte de l'analyse des récits de vie de Républicains espagnols, les différents camps constituent des lieux importants de l'analyse, à la fois par leur fréquence d'évocation dans les récits et les événements qui s'y rapportent. Par exemple, le camp situé à Argelès-sur-Mer admet plusieurs désignations qui varient selon les locuteurs et les moments du récit et rendent compte de ses différents usages : *camp de concentration*, *camp d'internement*, *camp de regroupement*, *le camp célèbre d'internement des Républicains espagnols* ; sa localisation peut être complète : *Argelès-sur-Mer* ou tronquée (Panckhurst, 2006) : *Argelès*.

L'identification d'un nom de lieu et des informations afférentes à ce lieu dans un texte a souvent pour objectif de représenter ces informations localisées c'est-à-dire de définir les objets cartographiques correspondant aux lieux et à leurs propriétés dans le texte, chaque objet cartographique étant caractérisé par son implantation, sa symbolisation et sa position (Bertin, 1967). Ce domaine de recherche est exploré à travers la représentation de récits de vie (Caquard, Cartwright, 2014) et (Olmedo, 2016), d'itinéraires de voyages (Santos *et al.*, 2017), de romans (Rosemberg, Trouin, 2017), de récits migratoires (Mekdjian, 2016), de films (Caquard, Fiset, 2013), d'analyses géopolitiques (Papin *et al.*, 2012), etc.

Positionner un lieu après avoir reconnu son identifiant suppose de disposer d'une ressource qui permette d'associer à chaque identifiant des coordonnées. La désambiguïsation (le choix entre, par exemple, les différentes entités géographiques ou administratives désignées par le nom *Paris* dans une base de données) est un problème bien connu en recherche automatique d'information et de nombreuses solutions heuristiques ont été apportées, par exemple (Cai, Tiang, 2016), (Yu *et al.*, 2016), (Blank, Henrich, 2016), (Leidner, Lieberman, 2011), (de Souza Da Silva, Ahlers, 2017).

La symbolisation i.e. les variables visuelles (Bertin, 1967) choisies pour traduire le message du concepteur de la carte se nourrit des propriétés des lieux identifiées dans les textes, et portées par les modifieurs des noms propres comme des noms communs de lieux. L'un des enjeux de la représentation cartographique est de proposer des variables visuelles et des variations de ces variables qui permettent de différencier visuellement un *espace pour la parole et l'action citoyennes* ou d'un *espace public agréable et pouvant profiter à tous*.

3 Expérimentations pour l'identification des noms de lieux

Dans le contexte du TAL, l'identification des noms de lieux bénéficie des méthodes et outils mis en place pour la recherche automatique des entités nommées, par exemple (Nouvel *et al.*, 2015),

qui exploitent des gazetiers et sont donc bien adaptées à la recherche des noms de lieux NPr. Mais ces outils et ces gazetiers sont inopérants pour identifier des lieux désignés par des noms communs pour lesquels d'autres méthodes ont dû être mises en œuvre. Une chaîne de traitements a été construite sous GATE[5] qui identifie NPr et Nc de lieux, et testée sur deux corpus (voir Table 1).

3.1 Corpus de travail

Le premier corpus (désormais CoRR) rassemble des récits de vie (transcrits manuellement, durée : 18h 30mn, 197 kmots) de Républicains espagnols ayant combattu puis fui le franquisme en Espagne, pour s'installer finalement en France entre 1936 et 1938. Le deuxième corpus (désormais CoMP) est constitué des contributions d'internautes sur différentes questions concernant l'aménagement de Paris ou de la région Ile-de-France, recueillies sur la plate-forme collaborative mise en place par la Mairie de Paris (1,4 Mmots). Ces deux corpus ont été annotés manuellement en lieux à l'aide d'un guide d'annotation inspiré des campagnes ESTER[6] et adapté pour tenir compte de la définition élargie des noms de lieux et des spécificités du corpus comme l'utilisation conjointe du français et de l'espagnol dans les récits de vie. L'extrait suivant provient du CoMP (les désignations des lieux sont en **gras** et les lieux NPr **gras et soulignés**) :

*Les **boulevards des maréchaux** marquent une **frontière** entre **Paris** et les **villes de la banlieue proche** et ne constituent actuellement que des **axes de circulation périphérique à Paris**. Les **trottoirs** et **contre-allées** qui les bordent sont larges mais ne sont absolument pas exploités, ni valorisés. Ils pourraient sûrement être mieux aménagés en y développant des **axes piétonniers végétalisés (promenades vertes)**, des **voies cyclistes**, des **zones de commerces** ou **lieux artistiques et culturels**, afin d'en faire de **véritables lieux de vie** unissant et bénéficiant aux **arrondissements périphériques de Paris** ainsi qu'aux **villes limitrophes**.*

3.2 Identification des lieux NPr et Nc

Les lieux NPr ont été identifiés à l'aide de dictionnaires construits *ad hoc* et de l'outil ANNIE de GATE qui reconnaît et annote dans des textes les occurrences des entrées des dictionnaires. Les dictionnaires ont été construits à partir de BDNyme[7], la base de données toponymique de l'IGN et la ressource collaborative GeoNames[8] qui propose à la fois endonyme(s) et exonymes, utiles pour CoRR où les lieux peuvent être désignés en français, espagnol ou catalan.

Concernant les lieux Nc, la méthode d'identification mise en place repose sur l'apprentissage automatique à partir d'extraits de corpus annotés manuellement. L'outil d'apprentissage automatique choisi est le Stanford Named Entity Recognizer (NER)[9] (Finkel *et al.*, 2005). Les deux corpus CoRR et CoMP ont été séparés en corpus d'apprentissage et corpus de validation et le Stanford NER a été entraîné sur le corpus d'apprentissage avant d'être intégré à une chaîne de traitements GATE construite *ad hoc*. Un lexique de mots génériques, géographiques ou

5 https://gate.ac.uk/

6 http://www.afcp-parole.org/camp_eval_systemes_transcription/docs/Conventions_EN_ESTER2_v01.pdf

7 http://www.professionnels.ign.fr/bdnyme

8 www.geonames.org

9 https://nlp.stanford.edu/software/CRF-NER.shtml

administratifs issus d'une ontologie des éléments du territoire (Mustière *et al.*, 2009), a été ajouté au moment de la construction du modèle d'apprentissage. Le lexique a aussi été enrichi avec des génériques plus spécifiques au corpus CoRR et concernant les lieux d'asile ou de transit comme : *camp de triage* ou *camp de concentration* et des objets localisés comme : *convoi* ou *train*. Enfin, différents paramétrages ont été expérimentés pour construire le modèle d'apprentissage. Les meilleurs résultats donnent une F-mesure à 0,67 (la précision est à 0,72 et le rappel à 0,63).

Ce modèle (Stanford NER entraîné sur un corpus d'apprentissage extrait de CoMP+CoRR) a été utilisé pour reconnaître les lieux Nc dans l'ensemble du corpus. Il permet, même si le niveau de la F-mesure est améliorable, de comparer la répartition entre lieux NPr et lieux Nc dans les différents corpus (cf. Table 1). Puisque le taux de rappel est faible, le nombre de lieux Nc est sous-estimé par rapport à l'annotation manuelle. Malgré cela, dans le corpus CoRR, le nombre de lieux Nc (49%) est quasi-égal au nombre de NPr (51%) ; dans le corpus CoMP, il est même très largement supérieur à celui de NPr (75% des noms de lieux sont désignés par des Nc).

corpus	#tokens	#lieux NPr	#lieux Nc	#Nc/ (#Nc+#NPr)	#lieux/ #tokens
CoRR	197 168	2 683	2 540	0,49	0,03
CoMP	1 416 753	17 513	52 213	0,75	0,07

TABLE 1 : Répartition des désignations de lieux NPr et Nc dans les différents corpus

4 Discussion

Dans le contexte du traitement automatique des langues, c'est la notion d'entité nommée spatiale (ENS) qui correspond à la notion de lieu. Cependant, la définition générale d'entité nommée que donne Ehrman (2008) « *on appelle entité nommée toute expression linguistique qui réfère à une entité unique du modèle de manière autonome dans le corpus* » reprend les caractéristiques principales des NPr purs : lien unique et stable entre désignation et objet référé, et écarte de fait certains lieux désignés par des Nc. Les expérimentations mises en place pour cartographier des récits de vie et des propositions pour l'aménagement de Paris et la région Ile-de-France démontrent à la fois la pertinence des lieux Nc et leur importance numérique dans ces textes. La localisation du texte qui passe par l'identification des noms de lieux ne peut donc se limiter à l'identification des NPr trouvés dans les gazetiers, quelle que soit la complétude des gazetiers. La définition apportée par Nouvel *et al.* (2015) qui considère comme des entités nommées les « *éléments informationnels pertinents dont on parle et qui jouent un rôle dans la description d'un évènement, d'un fait* » va dans ce sens puisqu'elle permet de retenir les lieux et les objets localisés désignés par des Nc. Cependant, adopter cette dernière définition revient à admettre que la caractérisation d'une ENS dépend du champ thématique dans lequel s'inscrit le texte, avec pour conséquence que les ressources nécessaires pour l'identification des ENS et des lieux élargis sont à adapter, voire à reconstruire, pour chaque nouveau corpus de travail ou question de recherche.

Les expérimentations ont montré aussi que le grain d'analyse qui correspond à la désignation des lieux par des NPr est parfois trop grossier pour atteindre les objectifs de cette analyse. Par exemple, pour identifier les souhaits des contributeurs sur l'aménagement de Paris, il ne suffit pas d'identifier les lieux de la capitale désignés par des NPr. Les propositions énoncées concernent des zones réduites (en **gras**) : *ré-activer les **rez-de-chaussée**, aménager un véritable espace public sur*

*les **marches***, *végétaliser la **façade*** qui s'organisent selon des relations de méronymie emboîtées, par exemple : *place de la Bastille/Opéra Bastille/**marches*** ou *place de la Bastille/Opéra Bastille/**façade*** ou *place de la Bastille/**zone centrale***.

S'intéresser aux Nc de lieu, et en particulier aux noms génériques qui peuvent constituer la tête de groupes nominaux (GN) désignant des lieux, permet d'identifier, en gommant la variété des NPr, les types de lieu évoqués dans les textes. Dans CoRR, les récits évoquent des lieux et objets localisés spécifiques à des parcours de migrants : *frontière (de Burlada+de contrôle+française +espagnole+de l'Italie+d'Espagne)*, *convoi (173 +de la montagne)*, *camp (de Gurs+d'Argelès+ de Saint-Cyprien)*. Dans CoMP, nombre de lieux désignés ne sont pertinents que par leurs propriétés qui sont décrites par les modifieurs. Ces lieux sont ainsi désignés par des GN dont les noms têtes sont vagues, et souvent interchangeables : *lieu+espace+endroit+zone+coin*. Et ce sont les modifieurs qui renseignent sur les demandes des Franciliens : *de lecture+de rencontres et de jeux+de co-working+végétalisé+public piéton+naturiste en centre-ville+naturiste public*.

Derrière une désignation NPr qui semble référer à un lieu précis de manière univoque et stable, le lieu référé peut être incertain, imprécis ou ambigu (de Runz, 2008 :25-37). Par exemple, Purves & Derungs (2015) montrent combien le terme *Alps* (*Alpes*), lorsqu'il s'agit de lui associer son emprise, désigne une zone vague dont la géométrie varie selon les locuteurs. En outre, le terme *Alps* n'est pas référencé dans SwissNames, le gazetier de l'agence nationale suisse de cartographie, tout comme *Alpes* n'est pas non plus référencé dans BDNyme. Les frontières, et donc les limites géométriques, de ces entités sont imprécises et incertaines. Ces questions de représentation sont au cœur de la démarche de la cartographie sensible qui, dans un contexte multidisciplinaire, vise à donner une représentation d'un espace vécu et perçu : cartographies autochtones du pays mapuche (Hirt, 2009), construction collaborative du récit du Grand Paris[10] (Rabie, 2017), développement rural et création du pays d'Auvergne (Lardon, Piveteau, 2005), représentation de l'ambiance sonore de la ville (Brayer, Laroche, 2016). De nouvelles solutions de représentation sont alors créées pour rendre compte de ces expériences perceptives des lieux, par exemple celles à base de textile (Olmedo, 2016) ou d'argile (Mekdjian, Amilhat Szary, 2015). Ces représentations insistent plus sur les relations entre les lieux et leur caractérisation (danger, faim, soins médicaux, peur, mort, etc.) dont les traces linguistiques sont fondées sur les modifieurs associés aux Nc de lieux, que sur leur localisation.

Les perspectives de ce travail reposent d'abord sur la publication du corpus patrimonial des récits de vie des Républicains espagnols en XML-TEI (entretiens oraux, transcriptions et annotations en lieux et sentiments) sur une infrastructure de recherche pour l'archivage pérenne des corpus textuels comme Ortolang ou CLARIN (*Common Language Resources and Technology Infrastructure*). L'identification des lieux Nc, par leur importance numérique dans les textes considérés, concourt à leur géo-localisation, et donc à leur représentation cartographique ; cependant la localisation de ces lieux Nc (*usine de textile*) et objets localisés (*banc*, *chêne centenaire*) qui s'appuie sur la localisation des lieux NPr est un problème difficile qui doit tenir compte des questions de portée et d'anaphore, en particulier pour les NPr de lieux. Enfin, les modifieurs contenus dans la désignation d'un lieu peuvent être mis en regard des caractéristiques visuelles de sa représentation. Une dernière perspective consisterait à approfondir cette comparaison, en montrant en particulier comment les informations sémantiques de la désignation d'un lieu pourraient alimenter les caractéristiques visuelles de sa représentation cartographique.

[10] http://mongrandparis.fr/

Références

AUBOUSSIER J. (2016), « De quoi Europe est-il le nom ? Enjeux et usages argumentatifs de la polyréférentialité », *Argumentation et Analyse du Discours* [En ligne], 17 | 2016, mis en ligne le 15 octobre 2016, Consulté le 11 décembre 2016.
URL : http://aad.revues.org/2216 ; [DOI : 10.4000/aad.2216]

BERTIN J. (1967). *Sémiologie Graphique. Les diagrammes, les réseaux, les cartes*, Paris, La Haye, Mouton, Gauthier-Villars, 1967. 2e édition : 1973, 3e édition : 1999, EHESS, Paris.

BLANK D., HENRICH A. *(2017)* A Depth-First Branch-and-Bound Algorithm for Geocoding Historic Itinerary Tables. Proceedings of the *10th Workshop on Geographic Information Retrieval, GIR 2016*, San Francisco, USA.

BLIDON M. (2008) « Jalons pour une géographie des homosexualités », *L'Espace géographique*, 2/2008 (Tome 37), p. 175-189. URL : http://www.cairn.info/revue-espace-geographique-2008-2-page-175.htm ; [DOI: 10.3917/eg.372.0175]

BRANDO C., DOMINGUES C., CAPEYRON M. (2016). Evaluation of NER systems for the recognition of place mentions in French thematic corpora. Proceedings of the *10th Workshop on Geographic Information Retrieval, GIR 2016*, San Francisco, USA.

BRAYER L., LAROCHE S. (2016). Représenter l'ambiance sonore de la ville. Retours sur un atelier pratique. Nicolas Rémy (dir.); Nicolas Tixier (dir.). *Ambiances, tomorrow. Proceedings of 3rd International Congress on Ambiances*, Volos, Greece, University of Thessaly, vol. 1, 271-276.

BRUNET R., FERRAS R., THERY H. (1993, première édition : 1992). *Les mots de la géographie, dictionnaire critique*. Collection Dynamiques du territoire, Reclus, La documentation française, Montpellier-Paris.

CAI G., TIAN Y. (2016). Towards Geo-referencing Infrastructure for Local News. Proceedings of the *10th Workshop on Geographic Information Retrieval, GIR 2016*, San Francisco, USA.

CALABRESE-STEIMBERG L. (2009) Nom propre et dénomination évènementielle : quelles différences en langue et en discours ? *Corela*. DOI: 10.4000/corela.173

CAQUARD S., CARTWRIGHT W. (2014). Narrative Cartography: From Mapping Stories to the Narrative of Maps and Mapping. *The Cartographic Journal*, Taylor & Francis, 51(2):101-106. DOI: 10.1179/0008704114Z.000000000130]

CAQUARD S., FISET J-P. (2014). How can we map stories? A cybercartographic application for narrative cartography. *Journal of Maps*, 10:1, 18-25, DOI:10.1080/17445647.2013.847387

CISLARU G. (2008). Le nom de pays comme outil de représentation sociale, *Mots. Les langages du politique* [En ligne], 86 | 2008, mis en ligne le 30 mars 2010, consulté le 13 août 2017. URL : http://mots.revues.org/13452

DUBOIS, J., MARCELLESI, J-B., MEVEL J-P., GIACOMO, M. (1994, nouvelle édition 2000). *Dictionnaire de linguistique et des sciences du langage*. Larousse.

EHRMANN M. (2008). *Les entités nommées, de la linguistique au TAL : statut théorique et méthodes de désambiguïsation*. Thèse de doctorat, Université Paris 7.

FINKEL, J. R., GRENAGER, T., MANNING, C. (2005). Incorporating non-local information into information extraction systems by gibbs sampling. In *Proceedings of the 43rd annual meeting on association for computational linguistics*, 363–370. Association for Computational Linguistics.

GAIO M., SALLABERRY C., NGUYEN V.T. (2012).Typage de noms toponymiques à des fins d'indexation géographique. *TAL 53(2),* 143-176

HIRT I. (2009). Cartographies autochtones. Eléments pour une analyse critique. *L'espace géographique*. 2009-2, 171-186.

JONASSON K. (1994) : *Le nom propre. Constructions et interprétations*, Louvain-la-Neuve, Duculot.

KLEIBER G. (1996). Noms propres et noms communs : un problème de dénomination. *Meta* 414 (1996): 567–589. DOI :10.7202/003323ar

LARDON S., PIVETEAU V. (2005). Méthodologie de diagnostic pour le projet de territoire : une approche par les modèles spatiaux », *Géocarrefour* [En ligne], vol. 80/2 | 2005, mis en ligne le 01 décembre 2008, consulté le 14 décembre 2016. URL : http://geocarrefour.revues.org/980 ; DOI : 10.4000/geocarrefour.980

LEIDNER J. L., LIEBERMAN. M. D. (2011). Detecting geographical references in the form of place names and associated spatial natural language. Actes de *SIGSPATIAL* Special, 3(2):5-11, July 2011.

LEROY S. (2004). *Le nom propre en français*. L'essentiel Français. Ophrys.

MEKDJIAN S. (2016). Les Récits Migratoires Sont-Ils Encore Possibles Dans le Domaine Des Refugee Studies? Analyse Critique et Expérimentation de Cartographies Créatives. *ACME: An International Journal for Critical Geographies***,** [S.l.], v. 15, n. 1, p. 150-186, mar. 2016. ISSN 1492-9732. Available at: <https://www.acme-journal.org/index.php/acme/article/view/1211/1168>. Date accessed: 07 feb. 2018.

MEKDJIAN S., AMILHAT SZARY A.-L. (2016). Cartographies traverses, des espaces où l'on ne finit jamais d'arriver. https://visionscarto.net/cartographies-traverses

MOLINO J. éd., 1982, « Le nom propre », *Langages*, n° 66.

MUSTIERE S., ABADIE N., AUSSENAC-GILLES N., BESSAGNET M.-N., KAMEL M., KERGOSIEN E., SAFAR B. (2009). GéOnto: Enrichissement d'une taxonomie de concepts topographiques. Proceedings of *Spatial Analysis and GEOmatics Sageo 2009*.

NOUVEL D., EHRMANN M., ROSSET S. (2015). *Les entités nommées pour le traitement automatique des langues*. ISTE editions.

OLMEDO E. (2016). « Femmes de Marrakech. Pour une cartographie émotionnelle des récits des femmes de Sidi Youssef Ben Ali, Marrakech, Maroc. », in FOURNIER M., *Cartographier les récits.* Clermont-Ferrand, Presses Universitaires Blaise Pascal, Coll. « À la croisée des SHS ».

PANCKHURST R (2006) : « Le discours électronique médié : bilan et perspectives », in A. PIOLAT (Éd.). *Lire, écrire, communiquer et apprendre avec Internet*, p. 345-366. Marseille, Éditions Solal.

PAPIN D., FATTORI F., HOLZINGER F. (2012). Cartographier des représentations géopolitiques. Cartographier les utopies. Journées d'études "Comment cartographier les récits documentaires et fictionnels ?", Clermont-Ferrand, 16-17 novembre 2012.

PAVEAU M-A. (2008). Le toponyme, désignateur souple et organisateur mémoriel. L'exemple du nom de bataille. *Mots. Les langages du politique* [En ligne], 86 | 2008, mis en ligne le 30 mars 2010, consulté le 02 octobre 2016. URL : http://mots.revues.org/13102 ; DOI : 10.4000/mots.13102

PURVES R. S., DERUNGS C. (2015). From Space to Place: Place-Based Explorations of Text. *International Journal of Humanities and Arts Computing*, Volume 9 Issue 1, 74-94, ISSN 1753-8548. https://doi.org/10.3366/ijhac.2015.0139

RABIE J. (2017). *Ce qui fait lieu. Vers une éthique chorographique*. Thèse en aménagement de l'espace et urbanisme. École doctorale Ville, Transports et Territoires/Lab'Urba/Université Paris-Est.

RECANATI F. (1983). La sémantique des noms propres : remarques sur la notion de « désignateur rigide». Langue française, 57, 106-118.

ROSEMBERG M., TROIN F. (2017). Cartographie du Marseille d'un héros de roman policier (*Total Khéops* de J.-C. Izzo). *M@ppemonde* 121, http://mappemonde.mgm.fr/121_as2/

DE RUNZ C. (2008). *Imperfection, temps et espace : modélisation, analyse et visualisation dans un SIG archéologique*. Thèse de doctorat, Université de Reims - Champagne Ardenne.

SANTOS R., MURRIETA-FLORES P., MARTINS B. (2017). An Automated Approach for Geocoding Tabular Itineraries. Proceedings of the *11th Workshop on Geographic Information Retrieval, GIR 2017*, Heidelberg, Germany.

DE SOUZA DA SILVA E., AHLERS D. (2017). Factorization Models for Spatiotemporal Retrieval. Proceedings of the *11th Workshop on Geographic Information Retrieval, GIR 2017*, Heidelberg, Germany.

VAN DE VELDE D. (2000) Existe-t-il des noms propres de temps ? *Lexique 15/Les noms propres : nature et détermination*, Septentrion, Presses universitaires.

YU L., LIU X, LI M, PENG P., LU F *(2016).* A Holistic Framework of Geographical Semantic Web Aligning. Proceedings of the *10th Workshop on Geographic Information Retrieval, GIR 2016*, San Francisco, USA.

JeuxDeLiens: Word Embeddings and Path-Based Similarity for Entity Linking using the French JeuxDeMots Lexical Semantic Network

Julien Plu[1] Kévin Cousot[2]
Mathieu Lafourcade[2] Raphaël Troncy[1]
Giuseppe Rizzo[3]
(1) EURECOM, Sophia-Antipolis, France
(2) LIRMM, Montpellier, France
(3) ISMB, Turin, Italy
julien.plu@eurecom.fr, kevin.cousot@lirmm.fr
mathieu.lafourcade@lirmm.fr, raphael.troncy@eurecom.fr
giuseppe.rizzo@ismb.it

RÉSUMÉ
Les systèmes de désambiguïsation d'entités nommées utilisent principalement sur des bases de connaissances encyclopédique telles que DBpedia ou Freebase. Dans ce papier, nous utilisons à la place, un réseau léxicalo-sémantique nommé JeuxDeMots pour conjointement désambiguïser et typer les entités nommées. Notre approche combine les plongements de mots et la similitude de chemins dans un graphe résultant à des résultats encourageants sur un ensemble de documents provenant du journal Le Monde.

ABSTRACT

Entity linking systems typically rely on encyclopedic knowledge bases such as DBpedia or Freebase. In this paper, we use, instead, a French lexical-semantic network named JeuxDeMots to jointly type and link entities. Our approach combines word embeddings and a path-based similarity resulting in encouraging results over a set of documents from the French Le Monde newspaper.

MOTS-CLÉS : entité nommée, désambiguïsation, jeuxdemots, réseau lexical.

KEYWORDS: named entity, disambiguation, jeuxdemots, lexical network.

1 Introduction

There is an exponential growth of textual content made available on the Web, produced by anyone via a broad diversity of publishing platforms. Automated solutions to extract actionable insights from this content is therefore of utmost importance. Focusing on textual content, we identified four main challenges that the NLP community is tackling for performing an entity linking task : dealing with different genres (social media, video subtitles, newswire articles, search queries), written in different languages, mentioning entities related to a variety of domains that can be classified with diverse sets of per-domain classes, and disambiguated against multiple kind of datasources (knowledge bases,

relational databases, lexical-semantic networks) (Plu, 2016).

Information extraction aims to get structured information from unstructured text by attempting to interpret natural language text to extracting information about entities, relations among entities and linking entities to external referents. In detail, entity recognition aims to locate and classify entities in text into defined classes such as Person, Location or Organization. Entity linking (or entity disambiguation) aims to disambiguate entities in text to their corresponding counterpart, referred as resource, contained in a knowledge base. Each resource represents a real world entity with a specific identifier.

Many knowledge bases can be used for doing entity linking : DBpedia (DBpedia, 2007), Freebase (Freebase, 2007), Wikidata (Wikidata, 2012) to name a few. Those knowledge bases are known for being broad in terms of coverage, while vertical knowledge bases also exist in specific domains, such as Geonames (Geonames, 2006), Musicbrainz (MusicBrainz, 2000), LinkedMDB (LinkedMDB, 2009), and here JeuxDeMots.

We have decided to focus our work on linking entities using a lexical-semantic network as a referent base. We have identified four main lexical-semantic networks for the French language : Wolf (Sagot & Fier, 2008), JeuxDeMots (Lafourcade, 2007b), FLN (Lux-Pogodalla & Polguère, 2011) and Babelnet (Navigli & Ponzetto, 2012), the latter being multilingual. We have selected the JeuxDeMots network (Lafourcade, 2007a) as it offers daily updates and it is well-suited for disambiguation thanks to its polysemy representation. This network is an oriented graph where the nodes can be labeled with terms, concepts or any kind of textual item. The edges are oriented, weighted and indicate a specific type of relation between two vertices. They can be lexical (lemma, locution, action to verb, etc.) or semantic (hypernymy, meronymy, agent, etc.). So far, the JeuxDeMots network has more than 1.2 million nodes, 67 million edges and about 100 relationship types.

The reminder of the paper is structured as follows : Section 2 detail the relevant previous work already done. Section 3 introduces our two steps approach for entity linking. Next, the Section 4 proposes an evaluation of this approach. Finally, we conclude and discuss some future work in Section 5.

2 Related Work

Currently, for the French language, the state of the art for entity linking suffers from a lack of evaluated approaches, resources and datasets. One of the most recent entity linking method (Stern *et al.*, 2012) relies on a statistical model trained on a manually annotated corpus and uses features computed using a basic heuristic tool and extracted from a large knowledge base. Unfortunately, this method is hard to reproduce since the knowledge base and the corpus used in the experiments are not publicly available. We describe two methods that can be applied to French : Babelfy (Moro *et al.*, 2014) and DBpedia Spotlight (Daiber *et al.*, 2013).

Babelfy proposes a graph-based approach where two main algorithms have been developed : random walk and a heuristic for finding the subgraph that contains most of the relations between the recognized mentions and candidates. The nodes are pairs (mention,entity) and the edges correspond to existing relationships in BabelNet (Navigli & Ponzetto, 2012) that are scored. Next, a semantic graph is built using word sense disambiguation (WSD) that extracts lexicographic concepts and does entity linking by matching strings with resources described in a knowledge base.

In contrast, DBpedia Spotlight relies on the so-called TF*ICF (Term Frequency-Inverse Candidate Frequency) score computed for each entity. The goal of this score is to show that the discriminative strength of a mention is inversely proportional to the number of candidates it is associated with. This means that a mention that commonly co-occurs with many candidates is less discriminative. Although, those methods can be applied on French documents, they have never been thoroughly evaluated with this language due to a lack of proper benchmark datasets. Next, those methods do not take into account the different semantics that an entity might have (see in Section 3 an example with *Paris*).

3 JeuxDeLiens Approach

We have originally developed a named entity recognition (NER) model trained with the ETAPE (Gravier *et al.*, 2012) corpus for recognizing entities in French documents. However, the corpus required a lot of pre-processing that was consisting of cleaning the transcripts (speech-to-text dataset). In fact, even after the cleaning the performance of the NER was very low. The extraction part not being our focus, we have assumed to take as input the surface forms of the entities that have to be linked and typed with the JeuxDeMots network.

Our entity linking approach is designed in two steps : *i)* word embeddings, and *ii)* path-based similarity. As a running example, we will use the following three sentences :

— Paris compte au 1er janvier 2013 plus de 2,2 millions d'habitants. [1] *Paris* stands for *the French capital*.

— En matière de commerce, Paris a clairement affiché sa volonté. [2] *Paris* stands for *the French government*.

— Paris fait ses premires apparitions comme mannequin dans plusieurs évènements de charité. [3] *Paris* here for *Paris Hilton*.

Entities are annotated beforehand in the text between double square brackets (e.g. [[Paris]]). We perform a preprocessing that consists in the following three steps :

1. tokenizing the input text by respecting the annotated entities between double square brackets ;
2. once the text is properly tokenized, we link each word to a node in the JeuxDeMots network, and we generate all the possible entity candidates. In the three sentences, the candidate entities for Paris are : *i)* Paris Hilton, *ii)* Paris, capitale [4], *iii)* Paris, gouvernement français [5], *iv)* Paris, prénom [6], and *v)* Paris, nom [7]. We also map the words that are multitokens expression such as *table de chevet* [8]. Finally, if no entity candidates are found, we assume the entity to be a novel entity [9] and will be linked to NIL ;
3. once the mapping is done, we remove the stopwords.

After this preprocessing, we have a list of words that represent the context of the document, and where each of them has its node mapped in the JeuxDeMots network. The next step is to select the best entity candidate in the list and we use a word2vec (Mikolov *et al.*, 2013) embedding.

1. As of 1 January 2013, Paris has more than 2.2 million inhabitants
2. In terms of trade, Paris has clearly stated its will.
3. Paris made her first appearances as a model in several charity events.
4. capital city
5. French government
6. firstname
7. familyname
8. nightstand
9. entities that do not appear in the data source being used

3.1 Word Embeddings

In order to properly disambiguate the entities, we use word2vec as word embeddings method with a model trained over the frWac corpus (Baroni *et al.*, 2009). This method has been chosen since a surface form might have more than one meaning as shown in our running example. A simple string comparison between surface forms, e.g. using the Levenshtein distance, is not efficient since the two contexts, of the entity and into JeuxDeMots shall be compared. In JeuxDeMots, the context of an entity is represented by his gloss (e.g. gouvernement français, prénom, capitale, etc.). For the first sentence, instead of comparing the surface form *Paris* with the words of the context, we use *gouvernement français* and the others gloss such as in Equation 1. The set W represents the following context : [compte, janvier, millions, habitants].

$$\begin{aligned} &w2v([gouvernement, francais], W) \\ &w2v([capitale], W) \\ &w2v([Paris, Hilton], W) \\ &w2v([prénom], W) \\ &w2v([nom], W) \end{aligned} \tag{1}$$

The entity candidate that got the best score is taken as the proper linked entity. We also propose to group the novel (*NIL*) entities that may identify the same real-world thing. We attach the same *NIL* value within and across documents. For example, if we take two different documents that share the same emergent entity, this entity will be linked to the same *NIL* value. We can then imagine different *NIL* values, such as *NIL_1*, *NIL_2*. We perform a string matching over the surface form between each novel entities that have been linked to *NIL* (or between each token if it is a multiple token mention). Concerning our running example, with this approach, all the *Paris* entities have been properly linked to their corresponding meaning. We use a path-based similarity method to find the type of the entity.

3.2 Path-Based Similarity

At this stage, we need to assign a type to an entity. From the JeuxDeMots network, we can gather the context words W, the entity e and the set of possible classes C. Although, we got W during the mapping step and e with word2vec, the class nodes are simply hand-picked nodes describing the class. Here, we only use *lieu* [10] for LOC, *organisation* for ORG and *personne* [11] for PER as classes.

The logic is to find the best path in the JeuxDeMots network from W to each element of C passing by e. An example is represented in Figure 1 for the first sentence. In case e is a novel entity, we try to directly find the best path from W to each element of C. The JeuxDeMots network has two characteristics that we have to take into account : *1)* a direct link from the entity to the class might not exist, there could be an indirect link (*Paris (capitale)* $\xrightarrow{\text{is-a}}$ métropole [12] $\xrightarrow{\text{is-a}}$ lieu), or multiple links (*Paris (capitale)* $\xrightarrow{\text{is-a}}$ {ville [13], préfecture [14], destination touristique [15]} and *2)* the JeuxDeMots taxonomy has no constraints, has multiple words to express the same meaning of a class and has loops.

10. place
11. person
12. metropolis
13. city
14. prefecture
15. touristic destination

In order to choose the class of the entities, we propose a graph similarity measure. Similarity measures are used in information retrieval (Franzoni *et al.*, 2014), semantic path analysis (Song *et al.*, 2015) and link prediction (Lü *et al.*, 2009). Some of them only take nodes attributes into account while others are based on nodes neighborhood or even paths (Liben-Nowell & Kleinberg, 2007). We propose a path-based similarity measure inspired from LP-index (Lü *et al.*, 2009). First, the score of a path μ is defined as :

$$score(\mu) = \delta^{|\mu|-1}. \sum_{\forall e=(u,v)\in\mu} w_e.|R^+_{T_e}(u)|^{-1} \tag{2}$$

With $\delta \in [0, 1]$ a length-malus, $|\mu|$ the length of the path, $R^+_{T_e}(u)$ the set of outgoing relations of u with the same type T_e as e and w_e its weight.

The similarity between two nodes u and v can then be expressed as :

$$sim(u, v) = \frac{1}{|M_{u,v}|}. \sum_{\forall \mu\in M_{u,v}} score(\mu) \tag{3}$$

With $M_{u,v}$ the set of paths between u and v. If the entity was found in the network we choose the class $c = \arg\max_{c\in C} sim_{ctx}(W, e, c)$ with :

$$sim_{ctx}(W, e, c) = sim_s(W, e) + sim(u, c) \tag{4}$$

And :

$$sim_s(W, e) = \frac{1}{|W|}. \sum_{\forall w\in W} sim(w, e) \tag{5}$$

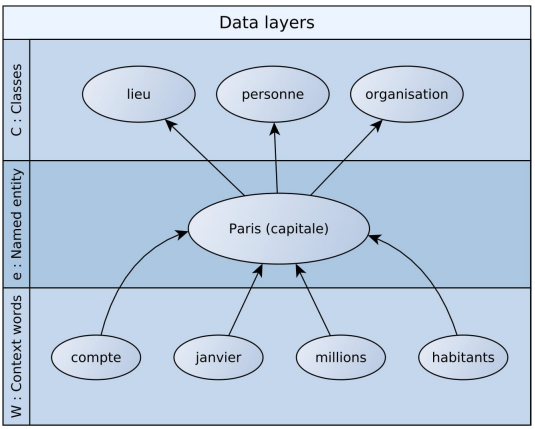


FIGURE 1 – The 3 data layers : classes, named entity and context words.

If the network does not contain the entity, then we only use its context to try and choose the class $c = \arg\max_{c\in C} sim_s(W, c)$. In practice, we limit the length of the paths to 2 and only the relevant types of relations are considered.

4 Evaluation

To evaluate JeuxDeLiens, we have selected 15 articles from the French newspaper Le Monde. We were not able to compare our approach with the ones used in Babelfly and DBpedia Spotlight. In fact, the used algorithms cannot be applied on the JeuxDeMots network (see Section 2) without changing their core.

The French community suffers from a lack of datasets for training entity linking systems. Therefore, we have decided to build a dataset based on Le Monde newspaper articles. We have randomly selected 15 articles that we have manually annotated. Only the entities that represent a *Person*, a *Location* or an *Organization* have been annotated, and linked to their JeuxDeMots ID if they exists in the lexical-semantic graph, otherwise they have been linked to NIL. The NIL entities keep the same ID across the documents, in order to respect the NIL clustering logic detailed in Section 3.1. The tool BRAT [16] has been used for the annotation task. In order to create a dataset with proper statistics, we have decided to align the numbers as much as possible over the ones from OKE Challenge 2015 Task 1 (Nuzzolese *et al.*, 2015) test dataset. We put together the statistics of these two datasets in Table 1 to be able to compare them.

As scorer, we use the neleval scorer (Hachey *et al.*, 2014) with the following metrics :

- *strong_typed_all_match* : performs a micro-averaged evaluation of all mentions. A mention is counted as correct if it is a correct link or a correct nil. A correct link must have the same span, entity type, and KB identifier as a gold link. A correct NIL must have the same span as a gold NIL.
- *entity_ceaf* : performs an evaluation based on a one-to-one alignment between system and gold entity clusters for both KB identifier and NIL across documents.
- *strong_typed_mention_match* : performs a micro-averaged evaluation of entity mentions. A system span must match a gold span exactly to be counted as correct and additionally requires the correct entity type.
- *strong_link_match* : performs a micro-averaged evaluation of links. A system link must have the same span and KB identifier as a gold link to be counted as correct.
- *strong_nil_match* : performs a micro-averaged evaluation of NIL entities. A system NIL must have the same span as a gold NIL to be counted as correct.
- *strong_all_match* : performs a micro-averaged link evaluation of all mentions. A mention is counted as correct if is either a link match or a NIL match as defined above.

As we can see, we evaluate JeuxDeLiens at multiple level in order to identify its strengths and weaknesses. The results are shown in Table 2. We also remind that we start from a score of 100% in F1 at extraction level as we take as input the correct surface forms of the entities.

	OKE2015	JeuxDeLiens
Number of unique entities of type PERSON	155	152
Number of unique entities of type LOCATION	90	102
Number of unique entities of type ORGANIZATION	122	131
Number of entities of type PERSON	317	228
Number of entities of type LOCATION	123	117
Number of entities of type ORGANIZATION	141	226

TABLE 1 – Statistics over the dataset

16. http://brat.nlplab.org/

	Precision	Recall	F1
strong_typed_all_match	63.2	63.2	63.2
entity_ceaf	73.3	93.3	82.1
strong_typed_mention_match	64.9	64.9	64.9
strong_link_match	72.9	89.7	80.5
strong_nil_match	100	50	66.7
strong_all_match	77.2	77.2	77.2

TABLE 2 – Results for JeuxDeLiens

From the error analysis conducted, it emerges that the heterogeneity of the network plays an important role for entity linking. Indeed, the knowledge is unevenly distributed across the network and while some domains are well-supplied, some are not. The entity coverage is very satisfying (89.7% recall for the *strong_link_match* score) but some entities have a very few incident edges making a path difficult to find. This is the case for *Boston Dynamics* and *BigDog*. Symmetrically, human-related nodes (*man*, *woman*, *human*...) are heavily well-supplied. As they are high-degree nodes, more paths are found leading to more misclassification favoring *PER*. This explains why some entities have been mistyped. The *entity_ceaf* score shows that the approach can succeed to give the same link for the entities that have the same meaning, including the ones that have been linked to same *NIL* across the documents. Nevertheless, the precision is a bit low because of a weakness of our *NIL* clustering method that might link unrelated entities if they share a same token in their mention. The score *strong_nil_match* reveals that when we link an entity to *NIL*, it is a good guess. However, our system still proposes some candidates for some entities that should be linked to *NIL*. Our word2vec measure has also a problem when processing a word that does not belong to the vocabulary since no similarity is computed. This impacts emergent entities that are present in JeuxDeMots but not in the frWac corpus.

5 Conclusion and Future Work

We have proposed JeuxDeLiens, a two steps method for disambiguating entities in French documents using the JeuxDeMots lexical semantic network. Although the evaluation has been made on a single newspaper article, the results of JeuxDeLiens are very encouraging and show that our approach can well disambiguate entities for French textual content. As future work, we aim to create a bigger evaluation dataset and to share it with the NLP community. We are also interested in finding a way to detect if no candidate entity matches, allowing us to spot missing data in the network. To do so, we plan to use the Deep Semantic Similarity Measure (DSSM) method (Huang *et al.*, 2013) in order to directly compute similarities between the nodes and identify if the entity belongs to the network or not. Finally, to solve the word2vec problem, we plan to use the FastText library (Bojanowski *et al.*, 2016) since it is robust against unknown vocabulary words because it uses letter-grams instead of tokens to compute embeddings. We can also improve the model by adding the full Wikipedia dump into the frWac corpus training data.

Références

BARONI M., BERNARDINI S., FERRARESI A. & ZANCHETTA E. (2009). The wacky wide web : a collection of very large linguistically processed web-crawled corpora. *Language Resources and Evaluation*.

BOJANOWSKI P., GRAVE E., JOULIN A. & MIKOLOV T. (2016). Enriching word vectors with subword information. *arXiv preprint arXiv :1607.04606*.

DAIBER J., JAKOB M., HOKAMP C. & MENDES P. N. (2013). Improving efficiency and accuracy in multilingual entity extraction. In *9th International Conference on Semantic Systems (I-SEMANTICS)*, Graz, Austria.

DBPEDIA (2007). Dbpedia. `http://wiki.dbpedia.org`.

FRANZONI V., MENCACCI M., MENGONI P. & MILANI A. (2014). Heuristics for semantic path search in wikipedia. In *14th International Conference on Computational Science and Its Applications (ICCSA)*.

FREEBASE (2007). Freebase. `https://www.freebase.com`.

GEONAMES (2006). Geonames. `http://www.geonames.org`.

GRAVIER G., ADDA G., PAULSSON N., CARR M., GIRAUDEL A. & GALIBERT O. (2012). The ETAPE corpus for the evaluation of speech-based TV content processing in the French language. In *8th International Conference on Language Resources and Evaluation (LREC)*.

HACHEY B., NOTHMAN J. & RADFORD W. (2014). Cheap and easy entity evaluation. In *52nd Annual Meeting of the Association for Computational Linguistics (ACL)*.

HUANG P.-S., HE X., GAO J., DENG L., ACERO A. & HECK L. (2013). Learning deep structured semantic models for web search using clickthrough data. In *22nd ACM International Conference on Information & Knowledge Management (CIKM)*.

LAFOURCADE M. (2007a). Jeux de mots. `www.jeuxdemots.org/`.

LAFOURCADE M. (2007b). Making people play for Lexical Acquisition with the JeuxDeMots prototype. In *7th International Symposium on Natural Language Processing (SNLP'07)*.

LIBEN-NOWELL D. & KLEINBERG J. (2007). The link-prediction problem for social networks. *Journal of the American Society for Information Science and Technology*.

LINKEDMDB (2009). Linkedmdb. `http://www.linkedmdb.org`.

LÜ L., JIN C.-H. & ZHOU T. (2009). Similarity index based on local paths for link prediction of complex networks. *Phys. Rev. E*.

LUX-POGODALLA V. & POLGUÈRE A. (2011). Construction of a french lexical network : Methodological issues.

MIKOLOV T., SUTSKEVER I., CHEN K., CORRADO G. & DEAN J. (2013). Distributed representations of words and phrases and their compositionality. *CoRR*.

MORO A., RAGANATO A. & NAVIGLI R. (2014). Entity linking meets word sense disambiguation : a unified approach. *TACL*.

MUSICBRAINZ (2000). Musicbrainz. `https://musicbrainz.org`.

NAVIGLI R. & PONZETTO S. P. (2012). Babelnet : The automatic construction, evaluation and application of a wide-coverage multilingual semantic network. *Artificial Intelligence*.

NUZZOLESE A., GENTILE A., PRESUTTI V., GANGEMI A., GARIGLIOTTI D. & NAVIGLI R. (2015). The 1st Open Knowledge Extraction Challenge. In *12th European Semantic Web Conference (ESWC)*.

PLU J. (2016). Knowledge Extraction in Web Media : At The Frontier of NLP, Machine Learning and Semantics. In *25th World Wide Web Conference (WWW), PhD Symposium*.

SAGOT B. & FIER D. (2008). Building a free french wordnet from multilingual resources. In *Ontolex 2008*.

SONG M., HEO G. & DING Y. (2015). Sempathfinder : Semantic path analysis for discovering publicly unknown knowledge. *Journal of Informetrics*.

STERN R., SAGOT B. & BÉCHET F. (2012). A joint named entity recognition and entity linking system. In *Proceedings of the Workshop on Innovative Hybrid Approaches to the Processing of Textual Data*.

WIKIDATA (2012). Wikidata. `https://www.wikidata.org`.

De l'usage réel des emojis à une prédiction de leurs catégories

Gaël Guibon[1, 2] Magalie Ochs[1, 3] Patrice Bellot
(1) LIS-CNRS UMR 7020, Aix-Marseille Université, 13000 France
(2) Caléa Solutions, 13002 France
prenom.nom@lis-lab.fr

RÉSUMÉ
L'utilisation des emojis dans les messageries sociales n'a eu de cesse d'augmenter ces dernières années. Plusieurs travaux récents ont porté sur la prédiction d'emojis afin d'épargner à l'utillisateur le parcours de librairies d'emojis de plus en plus conséquentes. Nous proposons une méthode permettant de récupérer automatiquement les catégories d'emojis à partir de leur contexte d'utilisation afin d'améliorer la prédiction finale. Pour ce faire nous utilisons des plongements lexicaux en considérant les emojis comme des mots présents dans des tweets. Nous appliquons ensuite un regroupement automatique restreint aux emojis visages afin de vérifier l'adéquation des résultats avec la théorie d'Ekman. L'approche est reproductible et applicable sur tous types d'emojis, ou lorsqu'il est nécessaire de prédire de nombreuses classes.

ABSTRACT
From Emoji Usage to Emoji-Category Prediction

Emoji usage drastically increased recently, they are becoming some of the most common ways to convey emotions and sentiments in social messaging applications. Several research works proposed to automatically recommend them to avoid having users scrolling down a library of thousands emojis. In order to improve emoji recommendation, we present a method to automatically extract emoji categories from their usage in tweets, following the assumption that emojis are part of written natual language, as words are. Thereby, emotion categories of face emojis were obtained directly from text in a fully reproductible way. These resources and methodology have multiple usages, including enhanced emoji understanding or enhanced emoji recommendation.

MOTS-CLÉS : emoji, recommandation, plongements lexicaux, ressource, regroupement.

KEYWORDS: emoji, recommendation, word embeddings, resource, clustering.

1 Introduction

Emoticônes (:-) , :P) et emojis (😍) sont deux manières de représenter des expressions du visage et peuvent être considérés comme substituts aux didascalies du domaine théâtral. Contrairement aux caractères que sont les émoticônes, les emojis sont des images qui peuvent représenter davantage que des expressions faciales, tels que des objets, concepts ou idées, et ont tendance à remplacer de plus en plus les émoticônes dans les messageries sociales (Pavalanathan & Eisenstein, 2015). Les 176 premiers emojis sont parus en 1999 par l'opérateur japonais NTT DOCOMO [1] avant d'être popularisés par leur intégration native dans le premier iPhone d'Apple, puis par Google et Samsung.

1. https://www.nttdocomo.co.jp/

Il y a désormais 2784 emojis standards [2].

Les emojis sont un sujet d'étude récent de plus en plus abordé depuis les trois dernières années. Dans les applications mobiles, les utilisateurs sont souvent confrontés à une librairie toujours plus grande d'emojis lorsqu'ils désirent n'en sélectionner qu'un seul. Une recommandation d'emojis apparaît donc nécessaire à l'amélioration de l'expérience utilisateur (*UX*), cependant plusieurs emojis peuvent fournir, à quelques nuances près, la même information ou la même émotion, comme c'est le cas pour 😀 et 😄. De plus, il n'est pas certain de savoir si les emojis peuvent être considérés en Traitement Automatique du Langage (TAL) comme de simples mots à part entière, ou comme des métadonnées greffées sur le contenu textuel. La majorité des systèmes de recommandation existants les considèrent comme des métadonnées, tandis que dans cet article nous considérons les emojis comme de simples mots, sans présupposés sémantiques.

Peu de systèmes de recommandation d'emojis ont été proposés jusqu'à présent, et la majorité ne cherchent à prédire qu'un seul emoji par message en mettant en place une prédiction multi-classe d'emojis. Avec 65% d'exactitude (Xie *et al.*, 2016) et 65% à 74% en moyenne harmonique (Barbieri *et al.*, 2017; Guibon *et al.*, 2017), les systèmes actuels restent perfectibles. D'autant plus qu'ils ne considèrent qu'un nombre limité d'emojis. C'est pourquoi dans cet article nous proposons une autre approche visant à extraire automatiquement des catégories d'emojis à partir de leur contexte d'utilisation réel, afin de substituer la prédiction d'emojis par une prédiction de catégories d'emojis. La création de catégories d'emojis a pour but d'étendre le choix proposé à l'utilisateur lors d'une prédiction de catégories, l'utilisateur est ensuite libre de sélectionner précisement l'emoji qui lui sied. Pour mettre en place cette méthodologie d'obtention de catégories d'emojis et vérifier son efficacité, nous nous focalisons dans un premier temps uniquement sur des emojis faciaux, afin de vérifier les catégories obtenues par une théorie existante de l'expression des émotions sur le visage, mais également parce qu'ils représentent un intérêt plus grand que de catégoriser des emojis objets (📸 📷).

Dans un premier temps, nous résumons l'état des avancées connexes au sujet (Section 2) avant de présenter les plongements lexicaux d'emojis mis en place (Section 3), ainsi que le regroupement automatique d'emojis basé sur ces derniers. Nous comparons ensuite théorie existante et résultats obtenus (Section 4) avant de conclure.

2 État de l'art

Les emojis ont récemment fait l'objet d'études dans différents domaines allant de la socio-linguistique à la classification par apprentissage automatique. En socio-linguistique, l'accent a été mis sur les divers usages d'emojis et la compréhension du message qui s'en voit alors facilitée dans 70% des cas (Kelly, 2015). Il a aussi été démontré que les emojis jouent plusieurs rôles dans une conversation (Kelly & Watts, 2015), rôles qui ne sont d'ailleurs pas nécessairement liés à l'expression des émotions : l'emoji peut alors assurer une fonction phatique, référentielle ou encore expressive, si l'on suit les six fonctions du schéma de Jakobson (Jakobson, 1960). Mettre en place une prédiction d'emojis efficace peut donc s'avérer déterminant dans l'expérience utilisateur.

La mise en place de cette prédiction d'emojis a suscité, quant à elle, des travaux de recherches aux objectifs variés et aux résultats perfectibles. Xie (Xie *et al.*, 2016) ont obtenu 65% d'exactitude à l'aide de *LSTM* hiérarchiques (Li *et al.*, 2015) pour la prédiction des trois emojis les plus usités

2. http://unicode.org/emoji/charts/full-emoji-list.html

dans des conversations Weibo [3]. Tandis que Barbieri (Barbieri *et al.*, 2017) ont obtenu 65% de moyenne harmonique en prédisant les 5 emojis les plus utilisés dans 40 millions de tweets à l'aide de *LSTM* (Hochreiter & Schmidhuber, 1997). Enfin, nous avons précédemment abordé l'approche par classification multi-étiquette sur des corpus de messages instantanés privés (Guibon *et al.*, 2017), obtenant 74% de moyenne harmonique sur 164 emojis possibles.

Il existe encore peu de ressources pour les emojis, la plupart étant en fait des modèles de plongements lexicaux appris. Ces derniers ont été étudiés récemment avec plusieurs approches quant à l'objet du plongement lexical, soit en considérant uniquement les descriptions Unicode des emojis comme groupe de métadonnées (Eisner *et al.*, 2016), soit en associant des significations et sens possibles aux descriptions (Ai *et al.*, 2017; Wijeratne *et al.*, 2017), ou alors en considérant directement les emojis en contexte dans des tweets. Nous abordons cette dernière approche qui a déjà fait l'objet d'un regroupement automatique avec un nombre arbitraire de groupes possibles (Barbieri *et al.*, 2016), ainsi que d'un regroupement hiérarchique (Pohl *et al.*, 2017), tous deux incluant tous types d'emojis.

Dans cet article, nous visons à obtenir automatiquement des groupes d'emojis à partir de leur usage réel sans pré-supposer un nombre de groupes possibles. Ce travail se rapproche donc de celui de Pohl (Pohl *et al.*, 2017) à la différence que nous proposons une méthodologie pour obtenir des groupes au sein d'un type spécifique d'emojis afin de les recommander à l'utilisateur par la suite.

3 Représentation vectorielle des emojis et plongements lexicaux

Si l'on en croit les métriques d'usage des emojis sur Twitter [4], les emojis les plus utilisés sont ceux représentant des émotions ou des sentiments, tels que 😂 💟 😍 ❤️ 😭. Nous cherchons donc à vérifier si l'usage des emojis faciaux suit implicitement une catégorisation des expressions du visage existante. Pour ce faire, nous observons l'usage de 63 emojis faciaux se rapprochant du visage humain, excluant ainsi les chats 😻, démons 😈, aliens 👽 ou autres 🙈. Ces 63 emojis ont été récupérés à partir de trois classes d'emojis présentes dans la classification Unicode : *face neutral*, *face positive* et *face negative*. Nous excluons les emojis très récents tel que 🤯 puisqu'ils sont absents de notre corpus.

Pour obtenir une répartition plus fine de ces emojis, nous mettons en place des plongements lexicaux d'emojis (dits *emoji embeddings*) sur un corpus de tweets.

Corpus de tweets. Notre corpus se compose de 695 031 tweets provenant du continent nord américain sur tous sujets, collectés à l'aide l'API de flux Twitter [5]. Pour nous assurer d'un corpus mono-lingue, tous ces tweets ont été préalablement filtrés par un détecteur de langue basé sur la liste des mots vides de NLTK [6] et leur ratio d'apparition dans le texte analysé.

Dans notre corpus nous considérons les emojis comme des mots comme les autres, bien qu'ils ne soient pas concernés par la lemmatisation effectuée avec WordNet (Miller, 1995). Les principales métriques du corpus sont disponibles au tableau 1.

Représentations vectorielles. Pour représenter les emojis nous avons utilisé *Word2Vec* (Rehurek & Sojka, 2010; Mikolov *et al.*, 2013b) implémenté dans Gensim [7] en variant deux architectures : Sac

3. http://www.weibo.com/
4. http://www.emojitracker.com/
5. https://dev.twitter.com/streaming/overview
6. http://www.nltk.org/
7. https://radimrehurek.com/gensim/models/word2vec.html

Tweets	695 031	**Emojis**	901 669
Mots/tweet	10,81 mots	**Emojis distincts**	844
Emoji/tweets		1,30	

TABLE 1 – Corpus de tweets contenant des emojis

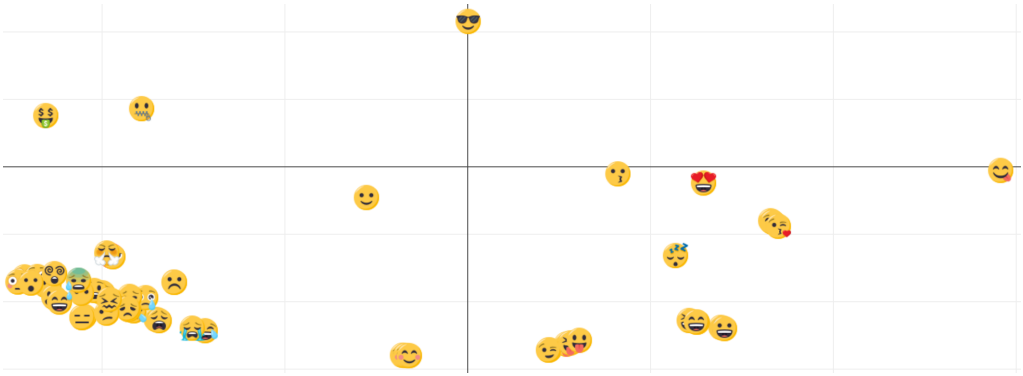


FIGURE 1 – Espace vectoriel de 63 emojis visages. Réduit à deux dimensions.

de mots continus (*CBOW*) pour prédire l'emoji à partir de son contexte avec l'algorithme *softmax* hiérarchique (Mikolov *et al.*, 2013a), et *Skip-Gram* pour prédire le contexte à partir de l'emoji. Les plongements lexicaux en *skip-gram* de Pohl (Pohl *et al.*, 2017) ont également été utilisés à titre de comparaison. Les vecteurs résultants de chacune des approches partagent une dimension de 300 en utilisant uniquement les mots appraissant au moins 5 fois. Ces modèles sont ensuite utilisés pour comparer leur impact sur la répartition automatique des emojis.

La répartition compacte des 63 emojis dans l'espace vectoriel est visible en Figure 1 après réduction de l'espace à 2 dimensions en utilisant l'algorithme TSNE [8] (Maaten & Hinton, 2008). La visualisation complète et intéractive sera disponible par lien et permet de voir s'y détacher plusieurs sous-groupes. Toutefois, les paramètres de TSNE ayant un fort impact sur la visualisation résultante, elle ne saurait à elle seule constituer une base fiable de répartition d'emojis. Dans la figure 1 les paramètres utilisés sont un taux d'apprentissage à 100, une perplexité à 30 et une *early exageration* de 2. Les autres paramètres étant ceux par défaut dans l'implémentation de Scikit-Learn [9].

Plusieurs approches sont possibles pour obtenir une répartition automatique d'emojis. Il est possible d'utiliser un seuil, arbitraire, sur la distance cosinus entre chaque élément de l'espace vectoriel, ou bien d'utiliser un algorithme de regroupement (*clustering*), ce que nous avons choisi de faire.

4 De l'utilisation réelle aux catégories d'emojis

Nous avons utilisé les vecteurs obtenus par plongements lexicaux comme données pour effectuer un regroupement automatique. Toutefois, notre objectif étant d'obtenir automatiquement des catégories d'emojis sans influencer le résultat par des pré-supposés, nous définissons le nombre de groupes

8. https://lvdmaaten.github.io/tsne/
9. http://scikit-learn.org/stable/modules/generated/sklearn.manifold.TSNE.html

(*clusters*) attendus comme étant égal au nombre d'éléments, soit au nombre d'emojis. À l'inverse des catégorisations automatiques d'emojis existantes telles que celle de Pohl (Pohl *et al.*, 2017) ou celle de Felbo (Felbo *et al.*, 2017), nous appliquons notre catégorisation uniquement sur les vecteurs d'emojis. Ces vecteurs étant obtenus en contexte général de mots et d'emojis, nous considérons l'information contextuelle comme déjà intégrée aux vecteurs qui font alors office de définition pour chaque emoji. L'ensemble des plongements lexicaux mélangeant termes et emojis est alors restreint avant catégorisation afin de ne l'appliquer que sur ce qui nous importe ici : les emojis et leurs relations.

Deux algorithmes courants ont été utilisés pour la catégorisation : les k moyens (*k-means*) (MacQueen *et al.*, 1967) et le regroupement spectral (*spectral clustering*) (Ng *et al.*, 2002). Le premier a été utilisé pour obtenir les centroïdes à partir des vecteurs d'emojis, considérant autant de centroïdes qu'il n'y a d'emojis afin de laisser la possibilité à chaque emoji de se retrouver isolé. Ajoutés à ces 63 groupes possibles 500 itérations avec 1000 initialisations, le résultat nous a donné 63 groupes dont plusieurs vides. En ignorant les groupes vides, nous obtenons 18 groupes contenant au moins un emoji. Le second, le regroupement spectral, a également été configuré avec 63 groupes possibles, un noyau gaussien au coefficient $\gamma 0.7$, un étiquetage discret et sans décomposition de matrice.

Au final, k moyens et regroupement spectral ont tous deux donné 18 groupes non-vides mais, malgré des résultats proches, nous avons opté pour le second puisqu'il ne divisait pas ou peu des groupes intuitifs tels que les emojis de bisous 😙😘.

Il convient de préciser qu'en utilisant des plongements lexicaux d'emojis en architecture *Skip-Gram*, nous avons obtenu 11 groupes à la granularité moins fine. Certains groupes obtenus de cette manière mélangent plusieurs emojis représentant des émotions différentes au sein d'un même groupe, telles que l'amour associée à la colère. Tandis que ceux obtenus avec une architecture *CBOW* sont plus cohérents, en plus de posséder une granularité plus fine. Ces derniers sont visibles au tableau 2.

Aucun	Aucun	Joie	Tristesse
💅	🤐	😙😘😗😚	😓😪😢😔😭😣😂😖😫
Joie	Honte	Excitation	Aucun (pas clair)
😆😃	😒😔	😋	😂😭
Colère	Amusement	Aucun	Plaisir sensoriel
😠😤👿	😅😩😛😝😜	🤑	😊😌☺😊
Peur / Surprise	Joie / Amusement	Satisfaction / Fierté	Aucun
😱😱😮😳😲😰😨😦	😄😆😅	😏😎😉	😩😕☹😟
Mépris		Excitation	
😑😶😐😒😕		😍	

TABLE 2 – Groupes d'emojis obtenus par regroupement spectral sur des plongements lexicaux en sac de mots continus. Les noms proviennent des catégories d'expressions de l'émotion d'Ekman.

Afin de ne pas nous baser uniquement sur nos propres plongements lexicaux, nous avons comparé nos résultats avec les groupes obtenus en utilisant les vecteurs d'emojis déjà appris par Pohl (Pohl *et al.*, 2017) en utilisant une architecture *Skip-gram*. Le regroupement appliqué est le même, la seule variable étant la différence de représentation vectorielle des emojis. Tout comme ce fut le cas avec nos plongements lexicaux appris en *Skip-gram*, les groupes obtenus furent plus généraux avec uniquement 6 groupes résultant pour une séparation large entre joie, colère, surprise et tristesse.

Considérant les groupes extraits à l'aide de modèles *Skip-gram* et ceux obtenus à l'aide de modèles *CBOW*, nous en sommes venus à la conclusion empirique que la seconde architecture est plus adaptée à un regroupement à granularité fine des emojis selon leur contexte. Ceci va à l'encontre des modèles

de plongements lexicaux existants qui se basent tous sur une architecture *Skip-gram* pour représenter les emojis en contexte, et démontre qu'elle est plus appropriée pour représenter les informations latentes issues de l'usage réel des emojis dans de courts messages instantanés.

Pour appuyer cette conclusion empirique, et parce que nous nous focalisons ici sur les emojis faciaux, nous avons comparé ces groupes à une théorie existante d'expressions faciales des émotions.

Validation théorique des groupes. L'ensemble des emojis traités étant restreint aux emojis représentant des expressions faciales, nous validons les résultats à l'aide de la théorie d'Ekman (Ekman, 1999) sur les 16 expressions basiques de l'émotion par le visage. Pour ce faire, chaque groupe voit ses éléments comparés aux émotions basiques d'Ekman. En assignant manuellement une catégorie d"Ekman à chaque emoji, puis en comparant les catégorisations manuelles et automatiques, nous obtenons une évaluation quantitative : 85,63% d'homogénéité, signifiant que la plupart des groupes contiennent uniquement des emojis d'une même catégorie ; et 69,45% de complétude, indiquant la capacité de tous les éléments d'une catégorie à se retrouver dans un même groupe. La moyenne harmonique des deux scores (*v-measure* (Rosenberg & Hirschberg, 2007)) est de 76,70%.

Les groupes et leur possible lien avec les émotions basiques sont visibles au tableau 2. De nombreux groupes correspondent aux catégories d'Ekman, bien que certains soient séparés en fonction de leur intensité comme c'est le cas pour la joie : une joie modérée 😄😃, et une autre plus intense 😄😆😁. La représentation reste toutefois imparfaite avec certaines catégories qui se chevauchent telles que la suprise et la peur, ou encore certains emojis non considérés par cette théorie qui se retrouvent alors isolés ou dans un groupe à part 🤑🤐🥳. Cet isolement maintient une certaine cohérence dans la catégorisation automatique des emojis à partir de leur contexte d'utilisation.

5 Conclusion et perspectives

Dans cet article nous avons mis en avant un travail en amont d'une recommandation d'emojis. Plutôt que de nous focaliser directement sur la recommandation d'emojis, nous avons voulu obtenir automatiquement des groupes d'emojis à partir de leur contexte d'utilisation réel, afin d'éviter de devoir les définir manuellement et de permettre d'obtenir des catégories adaptées à leur usage. Pour ce faire nous avons d'abord extrait les informations latentes relatives au contexte textuel de l'utilisation des emojis à l'aide de plongements lexicaux sur des tweets en considérant chaque emoji comme de simples mots. Puis, nous avons appliqué un regroupement automatique sur les vecteurs des emojis.

Pour valider notre approche nous avons limité la portée de l'expérience aux emojis représentant des expressions du visage afin de pouvoir comparer les catégories obtenues à une théorie existante. Nous avons ainsi remarqué que, contrairement aux modèles vectoriels d'emojis existants, l'architecture de sac de mots continus est plus apte à fournir des catégorisations fines des emojis, plus adéquate pour une recommandation ultérieure.

Les travaux présentés dans cet article représentent ainsi une méthodologie utile pour la recommandation d'emojis en préconisant de ne plus chercher à prédire les 2784 emojis existants comme c'est le cas actuellement dans l'état de l'art, mais prédire des catégories d'emojis obtenues automatiquement à l'aide de cette approche.

Références

AI W., LU X., LIU X., WANG N., HUANG G. & MEI Q. (2017). Untangling emoji popularity through semantic embeddings. In *ICWSM*, p. 2–11.

BARBIERI F., BALLESTEROS M. & SAGGION H. (2017). Are emojis predictable ? In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics : Volume 2, Short Papers*, volume 2, p. 105–111.

BARBIERI F., RONZANO F. & SAGGION H. (2016). What does this emoji mean ? a vector space skip-gram model for twitter emojis. In *Language Resources and Evaluation conference, LREC*, Portoroz, Slovenia.

EISNER B., ROCKTÄSCHEL T., AUGENSTEIN I., BOŠNJAK M. & RIEDEL S. (2016). emoji2vec : Learning emoji representations from their description. *arXiv preprint arXiv :1609.08359*.

EKMAN P. (1999). Basic emotions in t. dalgleish and t. power (eds.) the handbook of cognition and emotion pp. 45-60.

FELBO B., MISLOVE A., SØGAARD A., RAHWAN I. & LEHMANN S. (2017). Using millions of emoji occurrences to learn any-domain representations for detecting sentiment, emotion and sarcasm. *arXiv preprint arXiv :1708.00524*.

GUIBON G., OCHS M. & BELLOT P. (2017). Prédiction automatique d'emojis sentimentaux. In *COnférence en Recherche d'Information et Applications (CORIA)*.

HOCHREITER S. & SCHMIDHUBER J. (1997). Long short-term memory. *Neural Comput.*, **9**(8), 1735–1780.

JAKOBSON R. (1960). Closing statements : Linguistics and poetics, style in langage. *TA Sebeok, New York*.

KELLY C. (2015). Do you know what i mean > :(: A linguistic study of the understanding of emoticons and emojis in text messages.

KELLY R. & WATTS L. (2015). Characterising the inventive appropriation of emoji as relationally meaningful in mediated close personal relationships. *Experiences of Technology Appropriation : Unanticipated Users, Usage, Circumstances, and Design*.

LI J., LUONG M.-T. & JURAFSKY D. (2015). A hierarchical neural autoencoder for paragraphs and documents. *arXiv preprint arXiv :1506.01057*.

MAATEN L. V. D. & HINTON G. (2008). Visualizing data using t-sne. *Journal of Machine Learning Research*, **9**(Nov), 2579–2605.

MACQUEEN J. *et al.* (1967). Some methods for classification and analysis of multivariate observations. In *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability*, volume 1, p. 281–297 : Oakland, CA, USA.

MIKOLOV T., CHEN K., CORRADO G. & DEAN J. (2013a). Efficient estimation of word representations in vector space. *arXiv preprint arXiv :1301.3781*.

MIKOLOV T., SUTSKEVER I., CHEN K., CORRADO G. S. & DEAN J. (2013b). Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, p. 3111–3119.

MILLER G. A. (1995). Wordnet : a lexical database for english. *Communications of the ACM*, **38**(11), 39–41.

NG A. Y., JORDAN M. I. & WEISS Y. (2002). On spectral clustering : Analysis and an algorithm. In *Advances in neural information processing systems*, p. 849–856.

PAVALANATHAN U. & EISENSTEIN J. (2015). Emoticons vs. emojis on twitter : A causal inference approach. *arXiv preprint arXiv :1510.08480*.

POHL H., DOMIN C. & ROHS M. (2017). Beyond just text : Semantic emoji similarity modeling to support expressive communication. *ACM Transactions on Computer-Human Interaction (TOCHI)*, **24**(1), 6.

REHUREK R. & SOJKA P. (2010). Software framework for topic modelling with large corpora. In *In Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks* : Citeseer.

ROSENBERG A. & HIRSCHBERG J. (2007). V-measure : A conditional entropy-based external cluster evaluation measure. In *Proceedings of the 2007 joint conference on empirical methods in natural language processing and computational natural language learning (EMNLP-CoNLL)*.

WIJERATNE S., BALASURIYA L., SHETH A. P. & DORAN D. (2017). Emojinet : An open service and api for emoji sense discovery. In *ICWSM*, p. 437–447.

XIE R., LIU Z., YAN R. & SUN M. (2016). Neural emoji recommendation in dialogue systems. *arXiv preprint arXiv :1612.04609*.

Transfert de ressources sémantiques pour l'analyse de sentiments au niveau des aspects

Caroline Brun
Naver Labs Europe, 38240 Meylan, France
caroline.brun@naverlabs.com

RÉSUMÉ

Dans cet article, nous abordons le problème de la détection de la polarité pour l'analyse de sentiments au niveau des aspects dans un contexte bilingue : nous proposons d'adapter le composant de détection de polarité d'un système préexistant d'analyse de sentiments au niveau des aspects, très performant pour la tâche, et reposant sur l'utilisation de ressources sémantiques riches pour une langue donnée, à une langue sémantiquement moins richement dotée. L'idée sous-jacente est de réduire le besoin de supervision nécessaire à la construction des ressources sémantiques essentielles à notre système. À cette fin, la langue source, peu dotée, est traduite vers la langue cible, et les traductions parallèles sont ensuite alignées mot à mot. Les informations sémantiques riches sont alors extraites de la langue cible par le système de détection de polarité, et ces informations sont ensuite alignées vers la langue source. Nous présentons les différentes étapes de cette expérience, ainsi que l'évaluation finale. Nous concluons par quelques perspectives.

ABSTRACT

In this paper, we address the problem of automatic polarity detection in the context of Aspect Based Sentiment Analysis (ABSA), in a bilingual setting : we propose to adapt the polarity detection component of an existing high performing ABSA system, designed with rich semantic knowledge of a given language to a more poorly endowed language regarding semantics. The underlying idea is to minimize the level of supervision needed to develop the source language polarity component. To achieve this, we leverage on automatic translation of the ABSA source language into the ABSA target language, word alignment of these parallel translations, extraction of semantically-rich features on the ABSA target language, and back mapping of these target features onto the ABSA source language. The different steps of the design of the system are presented together with its evaluation. We conclude with some perspectives.

MOTS-CLÉS : Analyse de sentiments au niveau des aspects, ressources sémantiques, traduction.

KEYWORDS: Aspect-Based Sentiment Analysis, Semantic Resources, Translation.

1 Introduction

Les réseaux sociaux sont une source de grands volumes de contenus générés par les utilisateurs, dans lesquels les millions d'opinions exprimées par ces utilisateurs sont accessibles. Initialement, les travaux à dédiés l'analyse de sentiments se concentraient sur une analyse globale de la polarité d'un document. Cependant, les opinions sont rarement monodimensionnelles, mais le plus souvent

multi-dimensionnelles. L'analyse automatique des sentiments au niveau des aspects (Aspect Based Sentiment Analysis en anglais, ABSA), (Liu, 2012), (Ganu *et al.*, 2009), s'attelle précisément à extraire et résumer les opinions décrivant l'avis des utilisateurs sur des entités spécifiques et sur leurs aspects, c'est-à-dire les différentes caractéristiques qui les qualifient. Cette thématique, particulièrement attractive mais aussi complexe a été introduite dans le cadre du challenge SemEval en 2014 (Pontiki *et al.*, 2014), 2015 (Pontiki *et al.*, 2015) et 2016 (Pontiki *et al.*, 2016). Une des tâches du challenge consiste à analyser des revues phrase à phrase, afin de détecter les termes cibles du domaine sur lesquels une opinion est exprimée et de leur associer leurs aspects sémantiques et la polarité correspondante, en différentes langues et pour plusieurs domaines. Nous avons participé l'édition 2016 avec un système développé pour le français et l'anglais, et dont les résultats étaient très satisfaisants (Brun *et al.*, 2016). Dans le présent article, nous proposons de décrire une expérience visant à capitaliser sur le système préexistant de l'anglais, pour développer un composant de détection de la polarité pour une langue additionnelle, l'espagnol. Après la présentation de travaux connexes, nous décrivons la tâche du challenge SemEval2016 et le système que nous avons développé. Nous détaillons par la suite les différentes étapes de son adaptation à l'espagnol, ainsi que l'évaluation finale. Nous concluons par quelques perspectives de poursuite de ce travail.

2 Travaux connexes

La plupart des systèmes de détection des sentiments associés aux aspects utilisent des algorithmes d'apprentissage automatique tels que les "Support Vector Machines" (SVM), (Wagner *et al.*, 2014; Kiritchenko *et al.*, 2014), ou des "Conditionals Random Field" (CRF) (Toh & Wang, 2014; Hamdan *et al.*, 2015), souvent combinées à de l'information sémantique lexicale, des n-grams, des parties du discours (POS), et quelquefois de l'information syntactico-sémantique plus fine. Par exemple, la méthode proposée par (Kumar *et al.*, 2016) s'est avérée très performante sur les différentes langues du challenge SemEval2016. Le système inclut des informations extraites de graphes de dépendances et de thesaurus distributionnels appris pour les différentes langues et domaines, en plus d'informations lexicales dérivées de corpus non annotés. Les méthodes basées sur l'apprentissage profond commencent également à être appliquées avec succès à la tâche : une méthode basée sur les réseaux neuronaux convolutifs pour la classification en aspects et en polarité a obtenu des résultats intéressants pour plusieurs langues et domaines (Ruder *et al.*, 2016).

Un autre champ d'étude, l'analyse multilingue de sentiments, est également pertinent pour le présent article. En effet, certains travaux se sont concentrés sur l'adaptation de ressources associées aux sentiments, comme des lexiques, à partir de langues richement dotées (typiquement l'anglais) vers des langues moins dotées. L'article de (Mihalcea *et al.*, 2007) présente une méthode détectant la subjectivité dans des contextes multilingues en utilisant des lexiques bilingues et des projections cross-lingues. La méthode proposée par (Banea *et al.*, 2008) a permis d'obtenir des lexiques de subjectivité pour des langues autres que l'anglais en traduisant automatiquement le corpus MPQA [1], pour finalement obtenir des phrases annotées en roumain. En se basant sur ces travaux, (Wan, 2009) a proposé une méthode basée sur un algorithme de "co-training" pour analyser les sentiments de textes chinois et anglais en mode cross-lingue. En outre, (Balahur & Turchi, 2012) et (Balahur *et al.*, 2014) ont montré que, en utilisant des données d'entraînement obtenues par traduction automatique, ils pouvaient développer des classifieurs statistiques catégorisant la polarité des tweets en plusieurs langues. Ces différents travaux montrent que la traduction automatique se montre utile pour la classification

1. http ://mpqa.cs.pitt.edu/corpora/mpqa_corpus/

des sentiments pour des langues peu dotées en ressources lexico-sémantiques. Cependant, à notre connaissance, la traduction automatique n'a pas été utilisée dans le cadre de l'analyse de sentiments au niveau des aspects, ce qui est l'objet de l'expérience décrite dans le présent article.

3 Analyse de sentiments au niveau des aspects

L'analyse de sentiments au niveau des aspects est une tâche visant à capturer les sentiments exprimés sur différentes entités du monde, tels que des produits, des restaurants, des films, ou même des personnes, dans des contenus générés sur des médias sociaux. Les aspects sont des attributs de ces entités, par exemple le service dans un restaurant ou la qualité de l'écran d'un téléphone portable, généralement décrits sous forme d'une ontologie. Un système de détection des sentiments au niveau des aspects identifie l'entité sur laquelle une opinion s'exprime, qualifie la nature de l'aspect auquel il appartient et lui associe la polarité correspondante. La tâche d'analyse de sentiments au niveau des aspects de SemEval2016 (Pontiki *et al.*, 2016) était ainsi décomposée en trois sous-tâches : Phase A.1 : détection des termes sur lesquels une opinion s'exprime (terme cible) ; Phase A.2 : association de la catégorie sémantique de l'aspect associé au terme cible ; Phase B : association de la polarité (positif, négatif ou neutre) au couple <terme cible, catégorie de l'aspect>. Voici un exemple d'annotation pour l'anglais, dans le domaine des revues de restaurant :

<text>The food is fantastic and the waiting staff has been perfect.</text>
<Opinions>
<Opinion target="food" category="FOOD#QUALITY" polarity="positive" from="4" to="8"/>
<Opinion target="waiting staff" category="SERVICE#GENERAL" polarity="positive" from="31" to="44"/>
</Opinions>

Nous nous sommes intéressés au domaine des revues de restaurant, pour lequel les langues couvertes étaient l'anglais, le français, l'espagnol, le hollandais, le turque et le russe. Nous avons développé un système couvrant l'anglais et le français. Pour ce domaine, l'ontologie des aspects comprend 12 classes : *food#quality*, *food#style_options*, *food#prices*, *drink#quality*, *drink#style_options*, *drink#prices*, *service#general*, *ambience#general*, *location#general*, *restaurant#general*, *restaurant#prices* et *restaurant#misc* ; la polarité comprend 3 classes : positif, négatif, ou neutre. Nous nous intéressons ici au composant de détection de la polarité (phase B du challenge, pour laquelle l'annotation de référence en termes et en aspect est fournie), que nous transférons de l'anglais à l'espagnol. Le système initial que nous avons développé se base sur une méthode hybride combinant de l'information lexicale, syntaxique et sémantique à de la classification supervisée, (Brun *et al.*, 2016). Les traits linguistiques sont extraits par un analyseur syntaxique robuste et renseignent les différents algorithmes d'apprentissage. La détection des termes cibles des opinions est effectuée à l'aide d'un CRF (Lafferty *et al.*, 2001). La classification des aspects et des polarités associées aux termes sont réalisées par des modèles de régression logistique par ensemble. Une couche d'extraction de dépendances sémantiques dédiées à l'analyse de sentiments a été développée manuellement en aval de l'analyseur syntaxique, afin d'extraire des relations connectant prédicats de polarités et mots cibles des opinions, ce qui est illustré ci-dessous :

The food is fantastic and the waiting staff has been perfect.
Dépendances sémantiques : *sentim_posit(food,fantastic), sentim_posit(waiting staff, perfect).*

Ces dépendances sémantiques combinent les dépendances syntaxiques à des informations lexicales

concernant les schémas argumentaux, la polarité et les aspects. Elles sont utilisées comme traits pour la classification en polarité, et se montrent particulièrement discriminantes. Une validation croisée à 10 tours sur le corpus d'entraînement (environ 2000 phrases) a montré que leur introduction améliore d'environ 10 points l'exactitude de la classification en polarité. Lors de l'évaluation finale, le système s'est révèlé très performant pour cette sous-tâche de SemEval2016, puisqu'il obtient les meilleurs résultats pour le français et l'anglais. Nous reportons les résultats obtenus par notre système (NLE) sur les différentes sous-tâches dans le tableau 1.

	LANGUE : EN				LANGUE : FR			
Tâches	A.1 (F1)	A.2 (F1)	A.12 (F1)	B (Acc)	A.1 (F1)	A.2 (F1)	A.12 (F1)	B (Acc)
Baseline	59.93	44.07	37.80	76.48	52.61	45.45	33.02	67.4
NLE	68.70	61.98	48.89	**88.13**	**61.21**	65.32	**47.72**	**78.83**
Meilleur système	**73.03**	**72.34**	**52.61**	**88.13**	**61.21**	**66.67**	**47.72**	**78.83**

TABLE 1 – Résultats de Semeval2016 (domaine des restaurants) : A.1=détection des aspects, A.2=détection des termes, A1.2=détection des couples <terme,aspect>, B=détection des polarités

Cependant, si ces dépendances sémantiques sont discriminantes pour la classification en polarité, ce sont des ressources manuelles couteuses à développer, qui nécessitent un degré important de supervision. Leur adaptation à une autre langue s'avèrerait donc couteuse également. Dans la suite de cet article, nous présentons une expérience visant à automatiquement transférer les dépendances sémantiques de l'anglais vers une autre langue, l'espagnol, pour laquelle nous disposons uniquement d'un analyseur syntaxique générique mais pas de ressources lexicales de polarité et d'aspects, ni de dépendances sémantiques de sentiment.

4 Expérience de transfert bilingue de traits sémantiques

Nous présentons ici notre expérience de transfert de traits issus des dépendances sémantiques de l'anglais ($L_{absa-target}$) vers l'espagnol ($L_{absa-source}$). Pour cela, nous utilisation la traduction automatique du corpus d'entraînement de l'espagnol vers l'anglais, le composant d'extraction de traits linguistiques de l'anglais présenté dans la section précédente, et une méthode d'appariement de ces traits vers la représentation vectorielle de l'espagnol pour la classification en polarité. La méthode d'appariement se base sur l'alignement de mots phrase à phrase. La figure 1 décrit ce processus.

4.1 Traduction des corpus

Afin de traduire automatiquement les corpus d'entraînement et de test de l'espagnol vers l'anglais, nous utilisons des outils génériques de traduction automatique disponibles sur internet. Nous avons ainsi utilisé « Google Translate » [2] ainsi que l'API du traducteur de Microsoft [3], afin de comparer

2. https ://translate.google.com/
3. https ://www.microsoft.com/en-us/translator/translatorapi.aspx

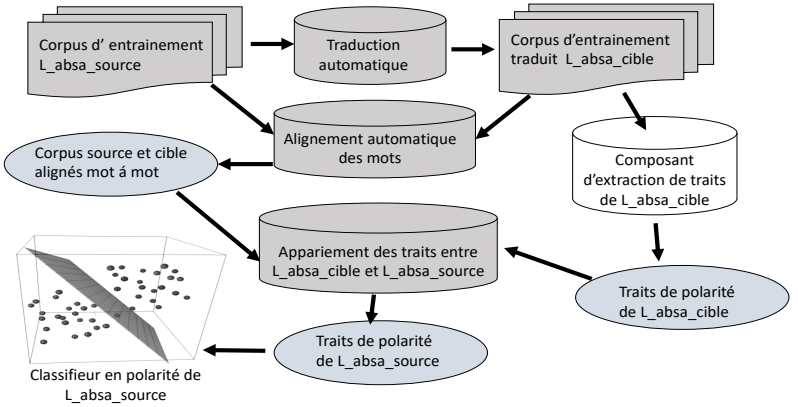


FIGURE 1 – Extraction des traits sémantiques pour $L_{absa-source}$

l'impact de la traduction sur notre méthode. Ces outils sont complexes à évaluer dans le contexte de notre expérience, puisque nous de disposons pas d'une traduction de référence, cependant nous les avons choisis car leur performances sont généralement très compétitives, (Isabelle *et al.*, 2017), (Jimeno Yepes *et al.*, 2017). La liste des phrases de l'espagnol $L_{absa-source}$ alignées avec leur traductions automatiques vers l'anglais est le résultat de cette étape.

4.2 Alignement des mots

A ce stade, le corpus d'entraînement de l'espagnol est aligné phrase à phrase avec sa traduction en anglais. Afin de pouvoir apparier les traits sémantiques qui vont être extraits de la traduction anglaise par notre composant d'extraction linguistique développé pour l'anglais, nous devons aligner mot à mot chacune des phrases de ces deux corpus parallèles. Plusieurs outils « open-source » d'alignement de mots à partir de corpus parallèles sont disponibles, par exemple les outils décrits dans (Och & Ney, 2003), ou encore (Germann, 2008). Nous avons choisi *fast align* (Dyer *et al.*, 2013) [4] pour sa simplicité d'utilisation. Cet outil prend en entrée des paires de phrases alignées et retourne les paires d'indices correspondant aux indices des mots alignés. Comme *fast align* est purement non supervisé, en plus des corpus parallèles dont nous disposons, nous lui avons également fourni des paires de mots du domaine (i.e. les revues de restaurants), c'est-à-dire des termes culinaires en espagnol et leur traductions anglaises, extraits de http ://www.linguasorb.com/Spanish/food-word-list).

4.3 Appariement des traits sémantiques

Dans un premier temps, le système développé pour l'anglais est appliqué sur le corpus parallèle anglais traduit du corpus d'entraînement de l'espagnol : ce système annote les termes cibles des

4. https ://github.com/clab/fast_align

opinions, leurs aspects et leurs polarités. Ce système utilise comme traits de décision les dépendances sémantiques que nous voulons transférer à l'espagnol. Ces traits peuvent être associés à des termes cibles des opinions ou à des phrases entières. D'autre part, le corpus d'entraînement de l'espagnol contient les annotations des termes cibles des opinions. L'alignement des mots présentés dans la section précédente nous permet d'aligner les termes cibles des opinions annotés dans le corpus d'entraînement espagnol avec les termes cibles des opinions extraites par notre système sur la traduction parallèle en anglais. L'appariement est réalisé en maximisant l'intersection des indices des mots contenus dans les termes source et cible. Suite à cet appariement des termes source et cible, les traits sémantiques associés aux termes anglais peuvent être directement transférés aux termes appariés en espagnol. Ils sont alors concaténés au vecteur de traits « baseline » dont nous disposons pour l'espagnol, lors de l'entraînement et de la phase de prédiction. Un exemple étape par étape est présenté sur le tableau 2.

espagnol	*El servicio es muy bueno y la calidad de la comida al mismo nivel*
traduction anglaise	*The service is very good and the quality of food at the same level*
alignement des mots	0-0 1-1 2-2 3-3 4-4 5-5 6-6 7-7 8-8 10-9 11-10 9-11 12-12 13-13 14-14
appariement des termes	<servicio, service>, <comida, food>
traits sémantiques anglais	sentim_posit(service, good) sentim_posit(food)
transfert vers l'espagnol	sentim_posit(servicio, bueno) sentim_posit(comida)

TABLE 2 – Exemple de transfert de traits

5 Evaluation

Le système développé pour l'espagnol et intégrant les traits transférés a été évalué selon le protocole et avec les données de l'espagnol de SemEval 2016. Le corpus d'entraînement comprend 2070 phrases et 2720 opinions exprimées (70.8% positives, 24.7% négatives et 4.5% neutres), le corpus de test comprend 881 phrases et 1072 opinions exprimées (69.9% positives, 25.7% négatives et 4.4% neutres). L'utilisation des traductions obtenues avec « Google Translate » (ABSA-GTR) et avec le traducteur de Microsoft (ABSA-MTR) permet d'évaluer l'impact de la traduction sur le système dans son ensemble. Un premier couple de « baselines » est également obtenu en appliquant le système complet développé pour l'anglais sur les traductions du jeu de test, puis en associant directement les polarités obtenues à la version espagnole via l'appariement des termes, pour les deux outils de traduction, ce qui donne les résultats de BL-GTR et BL-MTR. Une troisième résultat « baseline », ABSA-BL, applique directement le système minimal de l'espagnol en utilisant uniquement les traits standard (sac de mots, n-grams, dépendances syntaxiques générales). Les résultats en terme d'exactitude sont présentés sur le tableau 3, avec les résultats de référence de SemEval2016 (« baseline » : SE16-BL et meilleur système : SE16-BEST).

Système	BL-GTR	BL-MTR	ABSA-GTR	ABSA-MTR	ABSA-BL	SE16-BL	SE16-BEST
Exactitude	66.51	65.20	**84.23**	84.04	77.42	77.79	83.58

TABLE 3 – Exactitude des différents systèmes pour la détection de la polarité en espagnol

Les résultats basés sur les différents systèmes de traductions sont très proches, que ce soit pour les « baseline » ou les modèles utilisant le transfert de traits sémantique. Les résultats de ABSA-BL, système entraîné sur l'espagnol avec le jeu de traits standard, sont alignés avec la « baseline » de SemEval2016. Les « baselines » basées uniquement sur les modèle de l'anglais ont des résultats bien en deçà des autres systèmes, ce qui tend à justifier notre approche de transfert de traits. En effet, globalement le gain en performance obtenu par le transfert des traits sémantiques est important : les deux systèmes améliorent légèrement les résultats du meilleur système de SemEval2016 pour l'espagnol, (Kumar *et al.*, 2016), alors classé premier sur quatre participants. Les résultats détaillés par polarité pour ABSA-BL, ABSA-MTR et ABSA-GTR sont présentés sur le tableau 4.

	ABSA-BL			ABSA-MTR			ABSA-GTR		
Etiquette	P	R	F1	P	R	F1	P	R	F1
positive	0.78	**0.94**	0.85	0.85	**0.94**	0.89	**0.86**	**0.94**	**0.90**
négative	0.71	0.42	0.53	**0.79**	0.69	**0.74**	0.77	**0.70**	**0.74**
neutre	**0.75**	**0.06**	**0.11**	0.66	0.04	0.08	**0.75**	**0.06**	**0.11**

TABLE 4 – Résultats par polarité pour ABSA-BL, ABSA-MTR et ABSA-GTR

Les corpus étant fortement biaisés vers les opinions positives, le rappel et la précision sont importants pour cette étiquette. L'introduction via traduction de traits sémantiques relatifs à la polarité contribue donc significativement à l'amélioration des résultats, en particulier pour les polarités négatives. Nous prévoyons de réaliser une analyse des erreurs pour quantifier plus précisément, entre autres, celles qui sont dues à la qualité des traductions, à la qualité de l'alignement ou bien au système d'extraction de traits de l'anglais.

6 Conclusion et perspectives

Nous avons présenté dans cet article une expérience d'adaptation d'un système d'une langue (l'anglais), dotée de ressources sémantiques riches, vers une autre langue (l'espagnol), pour laquelle nous ne disposions a priori pas de ressources sémantiques dédiées à la tâche. À cette fin, nous avons développé une méthode de transfert de traits sémantiques de la langue richement dotée vers l'autre langue, méthode basée sur la traduction automatique du corpus annotés de la langue source vers la langue cible. Le système d'analyse de sentiments au niveau des aspects de la langue cible peut alors être appliqué afin d'extraire des traits sémantiques qui sont ensuite appariés par alignement puis transférés vers la langue source. Ces traits sont exploitables pour l'apprentissage du modèle d'annotation en polarité de la langue source par un pipeline similaire à celui de la langue initiale. L'évaluation des résultats est très prometteuse puisqu'elle montre une amélioration des meilleurs résultats de SemEval2016 pour l'espagnol. Par la suite, nous souhaitons appliquer une méthode similaire pour les autres langues de la tâche d'analyse de sentiments au niveau des aspects de SemEval16, à savoir le hollandais, le turque et le russe. Une autre perspective intéressante est l'investigation de méthodes de transfert d'apprentissage, comme celle proposée par (Zhou *et al.*, 2016) et de l'adapter à la tâche d'analyse de sentiments au niveau des aspects.

Références

BALAHUR A. & TURCHI M. (2012). Multilingual sentiment analysis using machine translation ? In *Proceedings of the 3rd Workshop in Computational Approaches to Subjectivity and Sentiment Analysis*, WASSA '12, p. 52–60, Stroudsburg, PA, USA : Association for Computational Linguistics.

BALAHUR A., TURCHI M., STEINBERGER R., PEREA-ORTEGA J. M., JACQUET G., KUCUK D., ZAVARELLA V. & GHALI A. E. (2014). Resource creation and evaluation for multilingual sentiment analysis in social media texts. In N. C. C. CHAIR), K. CHOUKRI, T. DECLERCK, H. LOFTSSON, B. MAEGAARD, J. MARIANI, A. MORENO, J. ODIJK & S. PIPERIDIS, Eds., *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, Reykjavik, Iceland : European Language Resources Association (ELRA).

BANEA C., MIHALCEA R. & WIEBE J. (2008). A bootstrapping method for building subjectivity lexicons for languages with scarce resources. In *Proceedings of the Sixth International Conference on Language Resources and Evaluation (LREC-08)*, Marrakech, Morocco : European Language Resources Association (ELRA). ACL Anthology Identifier : L08-1086.

BRUN C., PEREZ J. & ROUX C. (2016). XRCE at semeval-2016 task 5 : Feedbacked ensemble modeling on syntactico-semantic knowledge for aspect based sentiment analysis. In *Proceedings of the 10th International Workshop on Semantic Evaluation, SemEval@NAACL-HLT 2016, San Diego, CA, USA, June 16-17, 2016*, p. 277–281.

DYER C., CHAHUNEAU V. & SMITH N. A. (2013). A simple, fast, and effective reparameterization of ibm model 2. In *Proceedings of NAACL*, Atlanta : Association for Computational Linguistics.

GANU G., ELHADAD N. & MARIAN A. (2009). Beyond the stars : Improving rating predictions using review text content. In *Proceedings of the 12th International Workshop on the Web and Databases*, Providence, Rhode Island.

GERMANN U. (2008). Yawat : Yet another word alignment tool. In *Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics on Human Language Technologies : Demo Session*, HLT-Demonstrations '08, p. 20–23, Stroudsburg, PA, USA : Association for Computational Linguistics.

HAMDAN H., BELLOT P. & BECHET F. (2015). Lsislif : Crf and logistic regression for opinion target extraction and sentiment polarity analysis. In *Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015)*, p. 753–758, Denver, Colorado : Association for Computational Linguistics.

ISABELLE P., CHERRY C. & FOSTER G. F. (2017). A challenge set approach to evaluating machine translation. *CoRR*, **abs/1704.07431**.

JIMENO YEPES A., NEVEOL A., NEVES M., VERSPOOR K., BOJAR O., BOYER A., GROZEA C., HADDOW B., KITTNER M., LICHTBLAU Y., PECINA P., ROLLER R., ROSA R., SIU A., THOMAS P. & TRESCHER S. (2017). Findings of the wmt 2017 biomedical translation shared task. In *Proceedings of the Second Conference on Machine Translation*, p. 234–247, Copenhagen, Denmark : Association for Computational Linguistics.

KIRITCHENKO S., ZHU X., CHERRY C. & MOHAMMAD S. (2014). Nrc-canada-2014 : Detecting aspects and sentiment in customer reviews. In *Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014)*, p. 437–442, Dublin, Ireland : Association for Computational Linguistics and Dublin City University.

KUMAR A., KOHAIL S., KUMAR A., EKBAL A. & BIEMANN C. (2016). Iit-tuda at semeval-2016 task 5 : Beyond sentiment lexicon : Combining domain dependency and distributional semantics

features for aspect based sentiment analysis. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*, p. 1129–1135, San Diego, California : Association for Computational Linguistics.

LAFFERTY J. D., MCCALLUM A. & PEREIRA F. C. N. (2001). Conditional random fields : Probabilistic models for segmenting and labeling sequence data. In *Proceedings of the Eighteenth International Conference on Machine Learning*, ICML '01, p. 282–289, San Francisco, CA, USA : Morgan Kaufmann Publishers Inc.

LIU B. (2012). *Sentiment Analysis and Opinion Mining*. Synthesis Lectures on Human Language Technologies. Morgan & Claypool Publishers.

MIHALCEA R., BANEA C. & WIEBE J. (2007). Learning multilingual subjective language via cross-lingual projections. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, p. 976–983, Prague, Czech Republic : Association for Computational Linguistics.

OCH F. J. & NEY H. (2003). A systematic comparison of various statistical alignment models. *Computational Linguistics*, **29**(1), 19–51.

PONTIKI M., GALANIS D., PAPAGEORGIOU H., ANDROUTSOPOULOS I., MANANDHAR S., AL-SMADI M., AL-AYYOUB M., ZHAO Y., QIN B., CLERCQ O. D., HOSTE V., APIDIANAKI M., TANNIER X., LOUKACHEVITCH N., KOTELNIKOV E., BEL N., JIMÉNEZ-ZAFRA S. M. & ERYIĞIT G. (2016). SemEval-2016 task 5 : Aspect based sentiment analysis. In *Proceedings of the 10th International Workshop on Semantic Evaluation*, SemEval '16, San Diego, California : Association for Computational Linguistics.

PONTIKI M., GALANIS D., PAPAGEORGIOU H., MANANDHAR S. & ANDROUTSOPOULOS I. (2015). Semeval-2015 task 12 : Aspect based sentiment analysis. In *Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015)*, p. 486–495, Denver, Colorado : Association for Computational Linguistics.

PONTIKI M., GALANIS D., PAVLOPOULOS J., PAPAGEORGIOU H., ANDROUTSOPOULOS I. & MANANDHAR S. (2014). Semeval-2014 task 4 : Aspect based sentiment analysis. In *International Workshop on Semantic Evaluation (SemEval)*.

RUDER S., GHAFFARI P. & BRESLIN J. G. (2016). INSIGHT-1 at SemEval-2016 Task 5 : Deep Learning for Multilingual Aspect-based Sentiment Analysis. *ArXiv e-prints*.

TOH Z. & WANG W. (2014). Dlirec : Aspect term extraction and term polarity classification system. In *Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014)*, p. 235–240, Dublin, Ireland : Association for Computational Linguistics and Dublin City University.

WAGNER J., ARORA P., CORTES S., BARMAN U., BOGDANOVA D., FOSTER J. & TOUNSI L. (2014). Dcu : Aspect-based polarity classification for semeval task 4. In *Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014)*, p. 392–397, Dublin, Ireland : Association for Computational Linguistics and Dublin City University.

WAN X. (2009). Co-training for cross-lingual sentiment classification. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP : Volume 1 - Volume 1*, ACL '09, p. 235–243, Stroudsburg, PA, USA : Association for Computational Linguistics.

ZHOU G., ZENG Z., HUANG J. X. & HE T. (2016). Transfer learning for cross-lingual sentiment classification with weakly shared deep neural networks. In *Proceedings of the 39th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '16, p. 245–254, New York, NY, USA : ACM.

Apport des dépendances syntaxiques et des patrons séquentiels à l'extraction de relations

Kata Gábor[1], Nadège Lechevrel[2], Isabelle Tellier[3], Thierry Charnois[1], Haïfa Zargayouna[1], Davide Buscaldi[1]
(1) LIPN, CNRS (UMR 7030), Université Paris 13
(2) Université Paris-Ouest Nanterre La Défense
(3)LaTTiCe, CNRS (UMR 8094), ENS Paris, Université Sorbonne Nouvelle - Paris 3
PSL Research University, Université Sorbonne Paris Cité
utrucmuche@lab.fr, umachinchose@adresse-academique.fr

Résumé

Dans cet article, nous étudions la contribution de propriétés syntaxiques à la tâche de clustering d'instances de relations sémantiques. Les instances, constituées de couples de concepts apparaissant dans des textes scientifiques, sont représentées dans une matrice où on les croise avec une représentation de leur contexte de co-occurrence. Différentes variantes de représentations sont envisagées pour ce contexte, en faisant appel à la fouille de données séquentielles et à l'analyse syntaxique en dépendances. Nos comparaisons suggèrent que les attributs issus d'analyses syntaxiques permettent d'améliorer la qualité du clustering final.

Abstract

Integrating Dependency Parses with Sequential Patterns to Improve Relation Extraction

In this paper, we investigate the contribution of syntactic features to the task of unsupervised clustering of semantic relation instances. Instances, i.e. couples of concepts appearing in scientific texts, are represented in a couple-pattern matrix over co-occurrence contexts. Various possible contextual representation features are compared, using sequential pattern mining and syntactic path extraction. We compare the purely lexical feature space with a combined representation, and conclude that adding syntactic features has the potential to improve clustering performance.

Mots-clés : Extraction d'Information, relations sémantiques, apprentissage non supervisé.

Keywords: Information Extraction, semantic relations, clustering.

1 Introduction

La tâche d'extraction de relations vise à reconnaître automatiquement la nature des relations sémantiques qui relient des tuples d'entités ou de concepts présents dans un corpus. Cette tâche est une composante essentielle de l'extraction d'information et un préalable indispensable à l'alimentation automatique de bases de connaissances à partir de textes.

L'extraction de relations est le plus souvent abordée par apprentissage automatique supervisé,

en se servant en entraînement d'instances de tuples dont la relation sémantique est déjà connue et annotée (Hobbs & Riloff, 2010; Zhou *et al.*, 2005; Weeds *et al.*, 2014; Turney & Mohammad, 2014; Turney, 2012). Dans ce cas, le nombre et la nature des relations sémantiques possibles sont fixés à l'avance. L'extraction d'information en domain ouvert (OpenIE) (Banko *et al.*, 2007; Del Corro & Gemulla, 2013; Ferret, 2015) est beaucoup moins contrainte. Son but est d'inférer des relations sémantiques entre couples d'entités ou de concepts de manière non supervisée à partir des données textuelles quelconques. Certaines approches se focalisent sur l'extraction des instances de relations en contexte (Angeli *et al.*, 2015). Dans d'autres approches, les couples sont automatiquement regroupés dans des clusters, et la liste de leurs relations sémantiques possibles n'est pas fixée à l'avance. Cette méthode passe aussi par la représentation des instances dans un espace d'attributs. Mais, quand on opère de façon non supervisée, il est plus risqué d'utiliser un espace de représentation hétérogène pour les couples de concepts. C'est pourquoi, alors que les approches supervisées tentent de combiner toutes sortes d'attributs, les non supervisées exploitent en général essentiellement les segments de textes qui les relient (Hearst, 1992; Yangarber *et al.*, 2002; Béchet *et al.*, 2012; Turney, 2005, 2006). Les patrons ne se limitent pour autant pas nécessairement à de simples séquences de mots ; ils peuvent intégrer des combinaisons d'informations lexicales et syntaxiques (Fader *et al.*, 2011). D'autre part, il est également possible de construire l'espace d'attributs en s'appuyant non pas sur les patrons qui caractérisent les co-occurrences des deux entités, mais sur la représentation vectorielle des entités individuelles. De telles approches se sont montrées exploitables pour le calcul non supervisé des analogies relationnelles (Mikolov *et al.*, 2013), des relations lexicales comme l'hypéronymie (Santus *et al.*, 2014) et pour des relations sémantiques génériques (Gábor *et al.*, 2017), mais leur performance reste limitée sur les domaines de spécialité (Gábor *et al.*, 2016b).

C'est aussi dans un cadre non supervisé que nous nous plaçons. Nous ne cherchons toutefois pas à extraire des connaissances générales de textes quelconques, comme en OpenIE. Nous avons choisi de nous focaliser sur un corpus constitué d'articles du domaine du TAL, pour lesquels nous pouvons servir d'experts en validation. En ce sens, nos travaux se rapprochent de ceux menés dans le domaine des données médicales et de la bioinformatique, où l'analyse automatique d'articles scientifiques est déjà largement explorée. Or, dans ce domaine, il est courant d'intégrer dans les attributs le résultat d'analyseurs syntaxiques en dépendances (Fundel *et al.*, 2007; Porumb *et al.*, 2015). Le "plus court chemin syntaxique" en dépendances reliant deux entités ("shortest path hypothesis") est ainsi souvent utilisé (éventuellement en combinaison avec des méthodes à base de noyaux) pour représenter ce couple d'entités (Bunescu & Mooney, 2005; Mooney & Bunescu, 2005). Néanmoins, si cette approche a fait ses preuves en apprentissage supervisé et semi-supervisé (Nakamura-Delloye & Clergerie, 2010; Nakamura-Delloye & Stern, 2011), elle a été à notre connaissance encore peu explorée en clustering.

Dans une étude portant sur des articles de TAL, il a été récemment montré qu'une représentation à base de patrons séquentiels fréquents permettait un meilleur clustering des couples de concepts que la prise en compte du texte complet qui les sépare (Gábor *et al.*, 2016b). Nous proposons ici d'évaluer l'apport d'informations syntaxiques issues d'analyses en dépendances à cette tâche de clustering de couples de concepts scientifiques en domaine de spécialité.

Dans ce qui suit, nous décrivons tout d'abord (section 2) nos données et la nature de la tâche traitée. Les attributs syntaxiques utilisés sont présentés en section 3, tandis que la section 4 est consacrée aux conditions expérimentales de nos expériences et à leurs résultats.

La section 5 conclut en indiquant la direction de futures investigations.

2 Définition de la tâche

Soit a_1, a_2, b_1, b_2 des concepts extraits d'un corpus et pertinents pour le domaine considéré. a_1 et a_2 apparaissent dans une même phrase, de même que b_1 et b_2. Nous voulons réunir les couples $a = (a_1, a_2)$ et $b = (b_1, b_2)$ dans un même groupe (ou cluster) si les relations sémantiques qui lient a_1 à a_2 d'une part, et b_1 à b_2 d'autre part, sont similaires. Pour cela, nous avons besoin de représenter (a_1, a_2) et (b_1, b_2) dans un même espace vectoriel permettant de calculer la similarité $sim(a, b)$ sur laquelle se basera un algorithme de clustering.

Les conditions expérimentales et les données utilisées pour nos expériences sont les mêmes que celles décrites dans (Gábor *et al.*, 2016b), cela permettra les comparaisons. Le corpus initial est ACL-RelAcS (Gábor *et al.*, 2016a), issu du ACL Anthology Corpus (Radev *et al.*, 2009). Les concepts scientifiques pertinents pour faire partie des couples d'instances sont supposés connus. Les couples pris en compte sont ceux dont les concepts appartiennent à une même phrase. La Table 1 donne des exemples de différentes portions de textes reliant deux mêmes concepts dans le corpus.

argument 1	context/relation content	argument 2
domain task		*(language) data*
parser	*trained to minimize cost over*	sentences
learning algorithm	*is trained to simultaneously chunk*	sentences
parser	*is learned given a set of*	sentences
algorithm	*is presented for learning a*	phrase-structure

TABLE 1 – Exemple : relation "task_applied" sur un même couple de concepts dans différents contextes

Dans (Gábor *et al.*, 2016b), les attributs ayant permis les meilleurs clusterings étaient ceux extraits par fouille de données séquentielle (Srikant & Agrawal, 1996; Béchet *et al.*, 2012). Dans ce domaine, une séquence est une liste ordonnée de "littéraux" (ou items), dont un patron séquentiel est une sous-liste. Dans notre application, le rôle des littéraux est joué par les mots du corpus, auxquels aucun traitement linguistique n'a été appliqué. Seuls les patrons séquentiels "fermés", c'est-à-dire qui ne sont pas des sous-listes de patrons de même support, ont été considérés. Dans ce qui suit, nous cherchons à enrichir cet espace de nouveaux attributs en tenant compte de la syntaxe.

3 Represéntation syntaxique

Pour réaliser l'analyse syntaxique des phrases de notre corpus contenant des concepts, nous avons utilisé un parser en dépendances. Cette approche est la plus courante en extraction de relations (cf. l'état de l'art complet disponible dans (Valsamou, 2017)). Les analyses en dépendances via les parsers de type "shift-reduce" (Nivre, 2003) ne nécessitent pas la définition d'une grammaire formelle et sont plus adaptées aux langues à ordre souple. Dans

ce paradigme, la structure d'une phrase est décrite en termes de relations binaires typées entre mots (ou unités lexicales). Chaque mot est associé à un unique autre mot qui est son "gouverneur", dont il est soit un argument soit un modifieur. La Figure 1 montre une telle analyse pour une phrase du corpus.

Ayant choisi la représentation basique des dépendances universelles de Stanford, tous les arbres de dépendances obtenus sont des graphes orientés acycliques ou DAG en anglais (*Directed Acyclic Graphs*), même dans les cas de coordination. Les arcs sont étiquetés par la nature de la relation syntaxique qui relie un "gouverneur" à son "gouverné". Nous avons choisi d'utiliser l'analyseur syntaxique de Stanford version 3.8.0 car il offre une représentation en dépendances orientée vers la sémantique avec un jeu d'étiquettes proche du projet des Universal Dependencies qui anime une partie de la communauté TAL [1]. La version *collapsed* des dépendances n'a pas été utilisée dans cette expérience. Cette version produit des graphes orientés qui ne sont pas forcément des arbres. Cela permet de prendre en compte la variété des contextes, mais entraîne la possiblité d'avoir plus d'un plus court chemin entre deux entités. Or dans nos arbres, contrairement aux graphes généraux qui peuvent contenir un ou plusieurs chemins les plus courts, il n'y a qu'un seul plus court chemin possible.

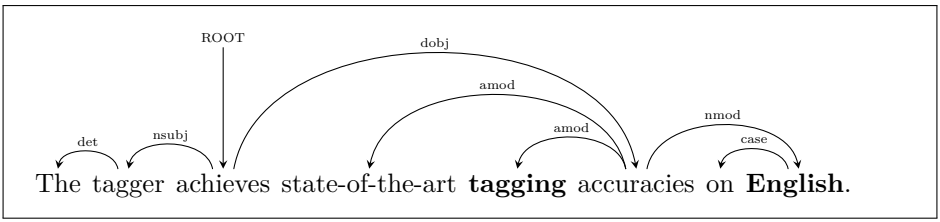


FIGURE 1 – Analyse syntaxique en dépendances d'une phrase du corpus

Bunescu et Mooney (Bunescu & Mooney, 2005) ont montré que le plus court chemin entre deux concepts ou entités dans une structure en dépendances est une information utile pour identifier la nature de la relation sémantique qui relie ces deux concepts. Dans l'analyse de la Figure 1, les deux mots "tagging" et "English" sont en gras parce que ce sont des concepts susceptibles de former un couple exprimant la relation sémantique *task_applied* (une tâche est appliquée sur des données) dont des exemples sont fournis en Table 1. Le plus court chemin qui les relie dans cette analyse passe par le mot "accuracies", dont ils sont tous les deux des modifieurs. Les informations syntaxiques que l'on peut extraire de l'analyse en dépendances pour caractériser ce couple de concepts sont détaillées en Figure 2.

1. Il s'agit d'un projet qui développe des corpus arborés pour de nombreuses langues dans le but de faciliter le développement d'analyseurs syntaxiques automatiques multilingues, l'apprentissage multilingue en ligne et les travaux de recherche en analyse syntaxique automatique dans une perspective de typologie linguistique. Le schéma d'annotation est hérité des dépendances de Stanford développées par de Marneffe et al. (en 2006, 2008, et 2014), du jeu d'étiquettes universelles de catégories *Part-of-speech* de Google et de l'*Interset interlingua for morphosyntactic tagsets*. Ici, nous nous référons uniquement aux *USD*, c'est-à-dire aux dépendances universelles de l'analyseur syntaxique de Stanford, et non au projet des *Universal Dependencies* (v2) dont on trouve un historique en ligne.

Analyse du plus court chemin entre les entités (tagging-5, English-8)
Path nodes: tagging-5, accuracies-6, english-8
Shortest path: **tagging** <– amod <– accuracies –> nmod –> **English**
Distance: 2

FIGURE 2 – Informations syntaxiques extraites de l'analyse pour un couple de concepts

4 Expériences

Nos expériences visent à mesurer l'impact de la prise en compte d'attributs syntaxiques sur les résultats du clustering de couples de concepts. Dans (Gábor *et al.*, 2016b,c), une classification manuelle des couples en 20 classes sémantiques avait été réalisée afin d'évaluer la qualité des clusterings. Nous reprenons ces données pour nos évaluations [2].

Pour les analyses syntaxiques, nous avons utilisé la version 3.8.0 du parser de Stanford, entraîné sur le Penn Treebank (Marneffe *et al.*, 2006). Les analyses en dépendances qu'il produit sont orientées vers la sémantique (Marneffe & Manning, 2008b). La version de base des types de dépendances (voir (Marneffe & Manning, 2008a) pour une description des étiquettes) a été choisie comme paramètre par défaut.

La sortie de l'analyseur syntaxique appliqué sur une phrase est une liste de relations binaires typées entre les mots de cette phrase. Pour chaque couple de concepts apparaissant dans une même phrase, nous avons extrait le plus court chemin en dépendances issu de ces relations binaires. La séquence des étiquettes de dépendances figurant sur ce plus court chemin (indépendamment de leur orientation) est devenu un attribut de l'espace de représentation des couples. Ainsi, l'exemple de la Figure 2 produit comme attribut la séquence "amod-nmod". Pour construire l'espace global de représentation des couples, le même filtre a été appliqué sur les séquences de mots et sur les séquences de dépendances : seuls les attributs apparaissant au moins 5 fois dans les données dont au moins 2 fois dans un couple en relation sémantique non vide ont été conservés. L'espace de représentation utilisé dans nos expériences était constitué soit des patrons séquentiels de mots seuls (1930 patrons séquentiels distincts), soit des seules séquences de dépendances (1191 séquences distinctes), soit d'un mélange des deux.

Le clustering a été réalisé par une approche hiérarchique agglomérative avec initialisation bissective (Zhao & Karypis, 2002) implementée dans cluto (Zhao *et al.*, 2005). L'initialisation par bissections successives produit un certain nombre de centroïdes qui viennent s'ajouter aux dimensions de l'espace de représentation initial. La valeur des couples sur ces nouvelles dimensions est leur distance au centroïde.

Les clusters obtenus ont été évalués relativement aux données de référence en termes de précision, rappel et F1-mesure. Les matrices auxquelles ont été soumis l'algorithme de clustering avec divers paramètres (3 valeurs distinctes du nombre de classes à trouver ont été testées) contenaient soit les simples comptes en nombre d'occurrences des attributs représentés dans chaque couple (Table 2), soit une variante avec les pondérations PPMI_α, fondées sur un calcul d'informations mutuelles (Levy *et al.*, 2015) (Table 3).

2. Les concepts exprimés par des expressions multimots ont été retirés de cette liste pour ces premières expériences, car ils peuvent poser des problèmes lors de l'analyse syntaxique. Nos données annotées comprennent

Input	#clusters	weight	Prec	Recall	F-measure
word seq	100	none	0.2590	0.1702	0.2054
word seq	50	none	0.2364	0.1873	0.2090
word seq	25	none	0.2293	0.2127	0.2207
parse	100	none	0.2760	0.1672	0.2082
parse	50	none	0.2443	0.1866	0.2116
parse	25	none	0.2346	0.2078	0.2204
combined	100	none	**0.2946**	0.1973	0.2364
combined	50	none	0.2755	0.2249	0.2476
combined	25	none	0.2640	**0.2390**	**0.2509**

TABLE 2 – Clustering : comparaison des différentes représentations sans pondération

Sans pondération, on observe des résultats très similaires en utilisant l'espace de représentation à base de patrons séquentiels de mots et celui à base de séquences de dépendances. Mais si on combine les deux représentations dans un seul espace, les résultats s'améliorent systématiquement d'environ 0,03 de F1-mesure. Cela suggère que les deux représentations capturent des informations différentes et complémentaires. Leur combinaison ne pénalise pas le clustering, malgré des différences de nature et d'échelles (les attributs syntaxiques sont plus génériques que les autres, et tendent donc à recevoir des valeurs plus grandes).

Features	#clusters	weight	Prec	Recall	F-measure
word seq	100	PPMI_α	**0.5337**	0.1906	0.2809
word seq	50	PPMI_α	0.4161	0.2759	0.3318
word seq	25	PPMI_α	0.3756	**0.2993**	**0.3332**
parse	100	PPMI_α	0.2300	0.1121	0.1507
parse	50	PPMI_α	0.1996	0.1461	0.1687
parse	25	PPMI_α	0.1952	0.1661	0.1795
combined	100	PPMI_α	0.4252	0.1738	0.2467
combined	50	PPMI_α	0.3552	0.2501	0.2935
combined	25	PPMI_α	0.3370	0.2889	0.3111

TABLE 3 – Clustering : comparaison des différentes représentations avec pondération PPMI_α

Un des résultats rapportés dans (Gábor *et al.*, 2016b) était que la pondération PPMI_α était particulièrement adaptée aux représentations à base de patrons séquentiels de mots. Nos expériences confirment ce constat, mais montrent aussi qu'il ne se généralise pas aux représentations à base de séquences de dépendances, pour lesquelles on note au contraire une dégradation. Nous espérions que la pondération permettrait d'équilibrer les échelles dans la variante combinée des réprésentations, et aboutirait aux meilleurs résultats, mais ce n'est pas confirmé par nos expériences. En observant de plus près les clusters obtenus, nous pensons que cette pondération est bénéfique à la prise en compte des patrons séquentiels peu fréquents, mais ne suffit pas à compenser les écarts d'échelles. Une explication possible est que l'information mutuelle redonne de l'importance aux événements rares (les patrons

donc un total de 567 couples, au lieu de 614 dans (Gábor *et al.*, 2016b).

séquentiels peu fréquents) mais que, pour ce qui concerne les attributs syntaxiques, ce sont les plus fréquents qui sont les plus porteurs d'information.

5 Conclusion et travaux futurs

Nous avons présenté dans cet article les premiers résultats d'expériences destinées à évaluer l'apport d'attributs syntaxiques pour le clustering de relations sémantiques dans un corpus de spécialité. Les résultats sont encourageants : une même relation sémantique peut être exprimée de façons très différentes dans un corpus, et il est donc important de prendre en compte des attributs divers susceptibles de la caractériser. Combiner des attributs de natures diverses est toujours risqué en clustering, nos premières expériences montrent que la combinaison envisagée est profitable. Il est aussi satisfaisant de constater que les seuls attributs syntaxiques permettent d'aboutir à d'aussi bons clusters que les seuls patrons séquentiels de mots.

La pondération PPMI_α, quant à elle, ne semble adaptée qu'à un seul type d'attributs parmi les deux familles testées. Cela ne remet pas en cause l'intérêt de la combinason (dans la Table 3, la combinaison des attributs reste meilleure que quand on utilise les attributs syntaxiques seuls) mais montre simplement que PPMI_α n'est pas adaptée aux dépendances syntaxiques. La suite de notre travail consistera donc notamment à chercher d'autres pondérations plus performantes pour cette représentation, et pour la combinaison des attributs.

Références

Angeli G., Premkumar M. J. J. & Manning C. D. (2015). Leveraging linguistic structure for open domain information extraction. In *ACL 2015*, p. 344–354.

Banko M., Cafarella J., Soderland S., Broadhead M. & Etzioni O. (2007). Open information extraction from the web. In *IJCAI*.

Béchet N., Cellier P., Charnois T. & Crémilleux B. (2012). Discovering linguistic patterns using sequence mining. In *CICLing '12*.

Bunescu R. C. & Mooney R. J. (2005). A shortest path dependency kernel for relation extraction. In *HLT-EMNLP'05*.

Del Corro L. & Gemulla R. (2013). Clausie : Clause-based open information extraction. In *International Conference on World Wide Web*, WWW '13.

Fader A., Soderland S. & Etzioni O. (2011). Identifying relations for open information extraction. In *EMNLP '11*.

Ferret O. (2015). *Language Production, Cognition, and the Lexicon*, chapter Typing Relations in Distributional Thesauri, p. 113–134. Springer International Publishing.

Fundel K., Küffner R. & Zimmer R. (2007). Relex—relation extraction using dependency parse trees. *Bioinformatics*, **23**(3), 365.

Gábor K., Zargayouna H., Buscaldi D., Tellier I. & Charnois T. (2016a). Semantic annotation of the acl anthology corpus for the automatic analysis of scientific literature. In *LREC '16*.

Gábor K., Zargayouna H., Buscaldi D., Tellier I. & Charnois T. (2016b). Unsupervised relation extraction in specialized corpora using sequence mining. In *Advances in Intelligent Data Analysis XV (IDA 2016), LNCS 9897*.

Gábor K., Zargayouna H., Tellier I., Buscaldi D. & Charnois T. (2016c). A typology of semantic relations dedicated to scientific literature analysis. In *SAVE-SD Workshop at the 25th World Wide Web Conference, LNCS 9792*.

Gábor K., Zargayouna H., Tellier I., Buscaldi D. & Charnois T. (2017). Exploring vector spaces for semantic relations. In *EMNLP 2017*, p. 1814–1823.

Hearst M. (1992). Automatic acquisition of hyponyms from large text corpora. In *COLING '92*, p. 539–545.

Hobbs J. R. & Riloff E. (2010). Information extraction. In N. Indurkhya & F. J. Damerau, Eds., *Handbook of Natural Language Processing, Second Edition*. Boca Raton, FL : CRC Press, Taylor and Francis Group.

Levy O., Goldberg Y. & Dagan I. (2015). Improving distributional similarity with lessons learned from word embeddings. *Transactions of the ACL*, **3**.

Marneffe M.-C. d., MacCartney B. & Manning C. D. (2006). Generating typed dependency parses from phrase structure parses. In *LREC '06*.

Marneffe M.-C. d. & Manning C. D. (2008a). Stanford typed dependencies manual. The Stanford NLP Group. revised for the Stanford Parser v. 3.7.0 in September 2016.

Marneffe M.-C. d. & Manning C. D. (2008b). The stanford typed dependencies representation. In *COLING Workshop on Cross-framework and Cross-domain Parser Evaluation*.

Mikolov T., Yih W. & Zweig G. (2013). Linguistic regularities in continuous space word representations. In *NAACL*.

Mooney R. J. & Bunescu R. (2005). Mining knowledge from text using information extraction. *SIGKDD Explor. Newsl.*, **7**(1), 3–10.

Nakamura-Delloye Y. & Clergerie E. d. l. (2010). Exploitation de résultats d'analyse syntaxique pour extraction semi-supervisée des chemins de relations. In *17e Conférence sur le Traitement Automatique des Langues Naturelles - TALN 2010*.

Nakamura-Delloye Y. & Stern R. (2011). Extraction de relations et de patrons de relations entre entités nommées en vue de l'enrichissement d'une ontologie. In *TOTh 2011 : Terminologie & Ontologie : Théories et Applications*, p.$\tilde{5}$0.

Nivre J. (2003). An efficient algorithm for projective dependency parsing. p. 149–160.

Porumb M., Barbantan I., Lemnaru C. & Potolea R. (2015). Remed : Automatic relation extraction from medical documents. In *Proceedings of the 17th International Conference on Information Integration and Web-based Applications & Services*, iiWAS '15 : ACM.

Radev D., Muthukrishnan P. & Qazvinian V. (2009). The ACL Anthology Network Corpus. In *ACL Workshop on Text and Citation Analysis for Scholarly Digital Libraries*.

Santus E., Lenci A., Lu Q. & Schulte Im Walde S. (2014). Chasing hypernyms in vector spaces with entropy. In *EACL'14*.

Srikant R. & Agrawal R. (1996). Mining sequential patterns : Generalizations and performance improvements. In *EDBT*, p. 3–17.

Turney P. D. (2005). Measuring semantic similarity by latent relational analysis. In *IJCAI-05*.

Turney P. D. (2006). Similarity of semantic relations. *CoRR*, **abs/cs/0608100**.

Turney P. D. (2012). Domain and function : A dual-space model of semantic relations and compositions. *Journal of Artificial Intelligence Research*, **44**.

Turney P. D. & Mohammad S. M. (2014). Experiments with three approaches to recognizing lexical entailment. *Natural Language Engineering*.

Valsamou D. (2017). *Extraction d'Information pour les réseaux de régulation de la graine chez Arabidopsis Thaliana*. Thèse de doctorat, Université Paris-Saclay, Ecole doctorale 580 Sciences et technologies de l'information et de la communication (STIC).

Weeds J., Clarke D., Reffin J., Weir D. & Keller B. (2014). Learning to distinguish hypernyms and co-hyponyms. In *COLING '14*.

Yangarber R., Lin W. & Grishman R. (2002). Unsupervised learning of generalized names. In *COLING '02*.

Zhao Y. & Karypis G. (2002). Evaluation of hierarchical clustering algorithms for document datasets. In *CIKM*.

Zhao Y., Karypis G. & Fayyad U. (2005). Hierarchical clustering algorithms for document datasets. *Data Mining for Knowledge Discovery*, **10**.

Zhou G., Su J., Zhang J. & Zhang M. (2005). Exploring various knowledge in relation extraction. In *ACL '05*.

Divergences entre annotations dans le projet *Universal Dependencies* et leur impact sur l'évaluation de l'étiquetage morpho-syntaxique

Guillaume Wisniewski François Yvon
LIMSI, CNRS, Univ. Paris-Sud, Université Paris-Saclay, 91 405 Orsay, France
prénom.nom@limsi.fr

RÉSUMÉ
Ce travail montre que la dégradation des performances souvent observée lors de l'application d'un analyseur morpho-syntaxique à des données hors domaine résulte souvent d'incohérences entre les annotations des ensembles de test et d'apprentissage. Nous montrons comment le principe de variation des annotations, introduit par Dickinson & Meurers (2003) pour identifier automatiquement les erreurs d'annotation, peut être utilisé pour identifier ces incohérences et évaluer leur impact sur les performances des analyseurs morpho-syntaxiques.

ABSTRACT
Evaluating Annotation Divergences in the UD Project

This work points out that the drop in performance often observed when applying a Part-of-Speech tagger to out-domain data may result from divergences between the annotation of train and test sets. We show how the annotation variation principle, introduced by (Dickinson & Meurers, 2003) to automatically identify errors in gold standard can be used to identify inconsistencies between annotations and to evaluate their impact on prediction performance.

MOTS-CLÉS : Erreur d'annotation, analyse morpho-syntaxique, adaptation au domaine.

KEYWORDS: Annotation error, PoS-tagging, domain adaptation.

1 Introduction

Les performances des analyseurs morpho-syntaxiques statistiques chutent de manière significative dès qu'ils sont utilisés pour étiqueter des phrases provenant d'un domaine différent de celui sur lequel ils ont été entrainés. Cette chute est généralement expliquée (Seddah *et al.*, 2012; Plank *et al.*, 2014; Bartenlian *et al.*, 2017) en supposant que la différence entre domaines se traduit par un changement de la distribution des vecteurs de caractéristiques $p(\mathbf{x})$ [1] liée à une variation dans l'usage des mots, à l'augmentation du nombre de mots hors-vocabulaire, à un usage différent de la ponctuation ou de la typographie, etc.

Cet article défend un point de vue différent et cherche à mettre en évidence qu'un autre facteur important pour expliquer la différence entre la entre domaines est le changement de la distribution jointe $p(\mathbf{x}, \mathbf{y})$, qui découle d'incohérences dans la manière dont les corpus sont étiquetés. En effet,

1. Cette situation est décrite dans la littérature statistique sous le nom de *covariate shift* (Shimodaira, 2000).

dans la quasi totalité des expériences d'adaptation au domaine, les ensembles de test et d'apprentissage ont été annotés de manière indépendante par des experts différents et même lorsqu'un même jeu d'étiquettes a été utilisé, les guides d'annotation ne sont pas toujours interprétés de la même manière. Le tableau 1 donne plusieurs exemples [2] de ces *divergences entre annotations*.

①	◇ Pour les particuliers , ce fut - et c' est toujours - une véritable aubaine , d' autant qu' en octobre 1989 , par peur d' une fuite de les placements dans une Europe bientôt sans frontières , le gouvernement socialiste (un comble !) exonéra pratiquement d'$_{\text{ADP}}$ impôts les revenus de les sicav monétaires , admises à le revenu de ces sicav à le voisinage de 10 % : ◇ Pour les particuliers , ce fut - et c' est toujours - une véritable aubaine , d' autant qu' en octobre 1989 , par peur d' une fuite de les placements dans une Europe bientôt sans frontières , le gouvernement socialiste (un comble !) exonéra pratiquement d'$_{\text{DET}}$ impôts les revenus de les sicav monétaires , admises à le revenu de ces sicav à le bénéfice de la capitalisation (voir ci - dessous) .
②	◇ Le Marché commun de le Golfe , devant déboucher vers la mise en place d' une monnaie commune , connaît aujourd'hui des retards en raison , entre autres$_{\text{NOUN}}$ de les divergences nées entre Ryadh et Abou Dhabi , principalement à le sujet de le siège de la future Banque$_{\text{PROPN}}$ centrale . ◇ Le Marché commun de le Golfe , devant déboucher vers la mise en place d' une monnaie commune , connaît aujourd'hui des retards en raison , entre autres$_{\text{PRON}}$ de les divergences nées entre Ryadh et Abou Dhabi , principalement à le sujet de le siège de la future Banque$_{\text{NOUN}}$ centrale .
③	◇ A quelques jours de le sommet de les sept grands pays industrialisés$_{\text{ADJ}}$, de le 6 à le 8 juillet à Munich , nous poursuivons la radioscopie de la situation économique de les pays riches (le monde de les 30 juin , 1 et 2 juillet) . ◇ A quelques jours de le sommet de les sept grands pays industrialisés$_{\text{VERB}}$, de le 6 à le 8 juillet à Munich , nous poursuivons notre enquête sur la situation de les pays riches (le monde de les 30 juin , 1 , 2 et 3 juillet) .

TABLE 1: Exemples de divergences entre les annotations des corpus français de l'UD : toutes ces phrases comportent une suite de mots en commun (les mots différents apparaissent dans une police plus claire) dont l'annotation est différente. Seules les étiquettes conflictuelles ont été représentées.

Dans cet article, nous décrivons plusieurs expériences qui mettent en évidence les divergences entre annotations au sein du projet *Universal Dependencies* (désormais UD)(Nivre *et al.*, 2017). Suivant l'approche développée par Boudin & Hernandez (2012), nous montrons comment le principe de variation d'annotations *(annotation variation principle)* introduit par Dickinson & Meurers (2003) peut être utilisé pour identifier les divergences entre annotations et mesurons l'impact de celles-ci sur l'évaluation des performances d'un analyseur morpho-syntaxique. Nos résultats suggèrent que la performance des étiqueteurs morpho-syntaxique sur des corpus hors domaine est souvent sous-estimée.

Le reste de cet article est organisé de la manière suivante. Nous commencerons par décrire les corpus et outils utilisés dans nos expériences (§2). Nous analyserons ensuite les résultats de deux expériences révélant des divergences d'annotations dans les corpus de l'UD (§3), avant de quantifier leur impact sur la qualité des prédictions (§4).

2 Cadre expérimental

Données Toutes les expériences présentées dans ce travail ont été réalisées avec les données du projet *Universal Dependencies* [3] (Nivre *et al.*, 2017). Ce projet a pour objectif de développer des

2. Les exemples sont présentés après application de pré-traitements, ce qui explique la présence de formes « décontractées » pour au (*à le*), et les espaces autour des symboles de ponctuation.

3. Nous avons utilisé la version 2.1 des données.

corpus étiquetés en PoS et en dépendances pour un large éventail de langues. La dernière version de l'UD rassemble 102 corpus couvrant 60 langues. Pour 22 langues, plusieurs corpus sont disponibles (jusqu'à 5 pour le français [4] et le tchèque) et il est possible de réaliser 63 expériences d'adaptation, chacune correspondant à l'apprentissage d'un étiqueteur sur un domaine pour ensuite l'exploiter sur un autre [5].

De nombreux corpus de l'UD résultent d'une conversion manuelle ou semi-automatique d'un corpus existant vers le schéma d'annotation de l'UD (Bosco *et al.*, 2013; Lipenkova & Souček, 2014). Dans la mesure où ces annotations ou conversions ont été réalisées indépendemment par différentes équipes, le risque d'incohérences et d'erreurs lors de l'interprétation des guides d'annotation est démultiplié ; plusieurs travaux (Vilares & Gómez-Rodríguez, 2017) ont ainsi récemment montré que de nombreux corpus d'une même langue ne sont pas étiqueté de manière cohérente. Un des principaux objectifs du présent article est de confirmer et de quantifier ces observations.

Analyseur morpho-syntaxique Nos expériences utilisent un analyseur morpho-syntaxique à base d'historique (Black *et al.*, 1992). Dans ces modèles, la prédiction d'une séquence d'étiquettes morpho-syntaxiques se réduit à une succession de problèmes de classification multi-classe : les étiquettes des mots de la phrase sont prédits l'une après l'autre par un perceptron moyenné. Nous utilisons un jeu de caractéristiques standard (Zhang & Nivre, 2011) : le mot courant, les mots suivants et précédents dans une fenêtre de taille deux, les deux dernières étiquettes prédites, etc. Ce modèle permet d'atteindre des performances proches de l'état de l'art tout en étant extrêmement rapide à entrainer, ce qui permet de multiplier les expériences.

Une description détaillée de ce modèle se trouve dans (Wisniewski *et al.*, 2014b,a).

Variation d'annotations Le principe de « variation d'annotations » (Boyd *et al.*, 2008) repose sur l'intuition que si deux séquences de mots identiques sont étiquetées de manière différente, il est fort probable qu'une de ces annotations contient une erreur. Dans ce travail, nous utiliserons ce principe pour détecter les divergences entre annotations de deux corpus.

Nous appelons *match* une séquence de mots qui apparait dans au moins deux phrases provenant de deux corpus différents et dont les annotations sont différentes. L'identification des matchs nécessite, dans un premier temps, de repérer toutes les séquences de mots identique dans deux corpus. Il s'agit d'une instance du problème de la plus longue sous-chaine répétée *(maximal repeat problem)* qui permet d'extraire efficacement (c.-à-d. avec une complexité en temps et en espace linéaire par rapport à longueur des deux corpus) tous les matchs à l'aide d'un arbre de suffixes généralisé (Gusfield, 1997).

Les matchs peuvent correspondre à des mots ou des groupes de mots qui sont effectivement ambigus, et pour lesquels la présence d'une divergence d'annotation est justifiée. Nous considérons deux heuristiques pour filtrer ces faux positifs. Tout d'abord, nous supposons que plus un match est long, plus il est vraisemblable qu'il résulte d'une incohérence ou d'une erreur d'annotation. Ainsi, le tableau 1 contient des matchs contenant plus de 10 mots qui sont tous imputables à des erreurs

4. Le projet contient notamment les conversions du *French Treebank* (Abeillé *et al.*, 2003) et du corpus Sequoia (Candito & Seddah, 2012) dans le formalisme UD ainsi que des corpus collectés spécifiquement comme ParTuT développé à l'université de Turin.

5. Dans 23 conditions, au moins un des corpus est trop petit pour entrainer un analyseur morpho-syntaxique et le corpus ne peut être utilisé que pour tester le modèle appris.

features	min.	max.	médiane	% best
words	66,9%	98,8%	89,7%	38,3%
labels	59,5%	98,9%	90,5%	48,3%
combi	67,0%	98,8%	90,0%	13,4%

TABLE 2: Précision moyenne obtenue sur les 63 conditions considérées par un classifieur cherchant à identifier à quel corpus appartient une phrase donnée.

d'annotation. Deuxièmement, avec l'heuristique *disjointe*, nous supposons qu'un match correspond à une ambiguïté naturelle lorsque les ensembles de ses étiquettes dans le premier corpus et dans le second corpus sont complètement disjoints et ne prenons pas ces cas en considération.

3 Divergence d'annotations dans les corpus de l'UD

L'objectif de cette section est de quantifier les divergences d'annotation dans l'UD. Au §3.1, nous utiliserons la divergence $\mathscr{H}$ pour caractériser les différences entre les corpus d'une même langue. Nous ferons ensuite le lien entre les erreurs de prédiction et les variations d'annotation (§3.2).

3.1 Caractérisation des différences entre corpus

Pour caractériser la différence entre deux corpus, nous utilisons la divergence $\mathscr{H}$ introduite par Ben-David *et al.* (2010) pour mesurer la similarité entre deux domaines (ou deux corpus). Cette mesure peut être estimée par le taux d'erreur d'un classifieur binaire entrainé pour décider si une phrase annotée provient du premier ou du second domaine. Intuitivement, plus ce taux d'erreur est élevé, plus il est difficile de discriminer les corpus d'apprentissage des corpus de test, et donc plus on peut penser qu'ils sont similaires.

Dans nos expériences, nous utilisons un modèle bayésien naïf [6] et trois ensembles de caractéristiques pour décrire une phrase et son annotation : *words*, pour lequel chaque exemple est représenté par un sac de mots (unigrammes et bigrammes) ; *labels*, dans lequel les exemples sont représentés de la même manière, mais en considérant cette fois, les PoS à la place des mots ; et *combi* qui utilise la même représentation après que les mots ont été concaténés avec leur PoS. Le premier jeu de caractéristiques permet d'identifier une différence dans la distribution des observations, les deux derniers des divergences d'annotation.

Le tableau 2 rapporte les résultats de cette expérience. Il indique, pour chaque jeu de caractéristiques considéré, pour quel pourcentage des 63 expériences d'adaptation possibles avec les corpus UD, ce jeu obtient le plus petit taux d'erreur. La figure 1 détaille ces résultats pour le français [7]. Les résultats sur les autres langues montrent des tendances similaires.

Il apparait que, dans la plupart des cas, il est possible d'identifier correctement le corpus d'où provient une phrase et son annotation. Bien que les chiffres bruts soient difficiles à interpréter (les

6. Nous avons utilisé l'implémentation fournie par (Pedregosa *et al.*, 2011) sans optimiser les hyper-paramètres.
7. Tous les scores sont moyennés sur 10 apprentissages et tests

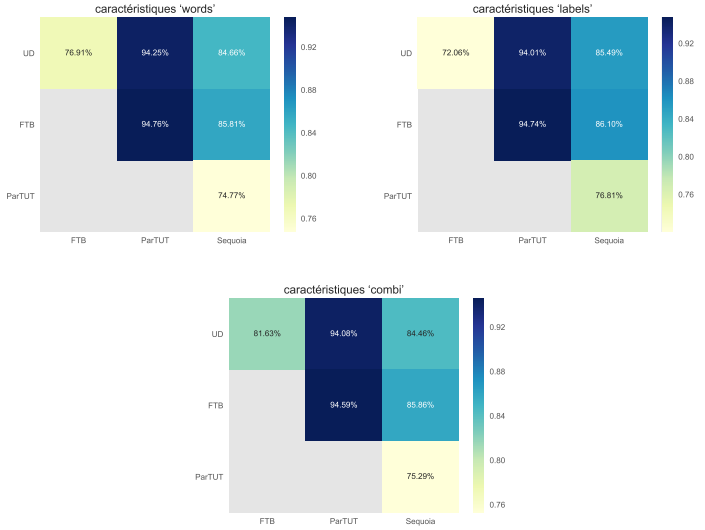


FIGURE 1: Précision obtenue par un classifieur cherchant à identifier à quel corpus appartient une phrase donnée sur les corpus français.

scores moyennés proviennent de nombreuses conditions expérimentales différentes, dont certaines correspondent à des problèmes de classification très déséquilibrés [8], ces résultats montrent clairement que, pour presque toutes les conditions, les caractéristiques les plus discriminantes incluent une description des étiquettes.

3.2 Impact des variations entre annotations sur la prédiction

Pour faire le lien entre erreurs de prédiction et divergences d'annotations, nous estimons le nombre de matchs dans lesquels au moins une étiquette n'est pas correctement prédite. En utilisant l'heuristique « disjointe » pour filtrer les matchs, il apparait que 70,2% (resp. 73,0%) des correspondances pour l'anglais (resp. français) contiennent une erreur de prédiction. Ces chiffres tombent à 51,7% (resp. 49,9%) lorsque les matchs ne sont pas filtrés et contiennent donc plus de mots ambigus. La figure 2 montre que filtrer les matchs selon leur longueur conduit à un effet similaire.

Toutes ces observations suggèrent que les variations entre annotations donnent souvent lieu à des erreurs de prédiction, surtout lorsqu'il est probable que la variation résulte d'une erreur d'annotation.

4 Évaluation hors domaine d'un analyseur morpho-syntaxique

Pour évaluer l'impact des erreurs d'annotation sur la qualité des prédictions d'un analyseur morpho-syntaxique, nous proposons de comparer $\varepsilon_{\text{full}}$, le taux d'erreur d'un analyseur estimé sur l'ensemble

8. Le rapport entre le nombre d'exemples de chaque corpus peut atteindre 88.

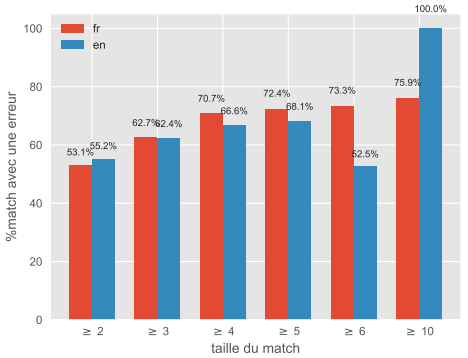


FIGURE 2: Matchs contenant au moins une étiquette mal prédite en fonction de la taille du match. Selon les jeux de données, le taux d'erreur sur l'ensemble du corpus est compris entre 5% et 15%.

du corpus de test et $\varepsilon_{\text{ignoring}}$ le taux d'erreur estimé en ignorant les matchs. Plus précisément, $\varepsilon_{\text{ignoring}}$ est défini de la manière suivante :

$$\varepsilon_{\text{ignoring}} = \frac{\#\{\text{errors}\} - \#\{\text{errors in matches}\}}{\#\{\text{words}\}} \quad (1)$$

où $\#\{\text{errors in matches}\}$ est le nombre d'erreurs dans des matchs, après filtrage. Intuitivement, ce taux d'erreur correspond à un taux d'erreur « oracle » qui serait obtenu si l'analyseur morpho-syntaxique prédisait correctement toutes les étiquettes des matchs.

La figure 3 représente ces taux d'erreur pour les corpus français lorsque sont ignorés les matchs ne respectant pas l'heuristique disjointe, les matchs de trois mots et plus, enfin tous les matchs. Supposer que les étiquettes des matchs sont correctement prédites permet de réduire de manière importante le taux d'erreur, au point que $\varepsilon_{\text{ignoring}}$ est souvent du même ordre de grandeur que le taux d'erreur obtenu sur un corpus du même domaine. En fait, dans plus de 43% (resp. 25%) des conditions, le taux d'erreur obtenu en ignorant les erreurs dans les matchs filtrés avec l'heuristique disjointe (resp. en fonction de la longueur du match) est inférieur au taux d'erreur obtenu sur les données du domaine.

Ces taux d'erreur sont naturellement sous-estimés puisque nous avons supprimé, dans ces expériences, des mots ou des structures qui étaient ambigus et dont l'étiquette est donc plus difficile à prédire. Ils peuvent toutefois être considérés comme une valeur oracle de la qualité de la prédiction.

Pour évaluer la qualité des différentes heuristiques de filtrage considérée, nous avons manuellement vérifié tous les matchs entre le corpus d'apprentissage `UD_French` et le corpus de test `FTB_French` et corrigé les différentes incohérences et erreur d'annotation (près de 2 000 étiquettes ont été modifiées). Lorsqu'il est entrainé sur le corpus d'origine, un analyseur morpho-syntaxique atteint un taux d'erreur de 6,8% sur le corpus FTB et 4,5% sur le corpus de test du même domaine. Après correction, le taux d'erreur hors-domaine tombe à 5,1%. Cette valeur est proche du taux d'erreur obtenu en ignorant les matchs de trois mots et plus, une observation qui valide l'heuristique considérée.

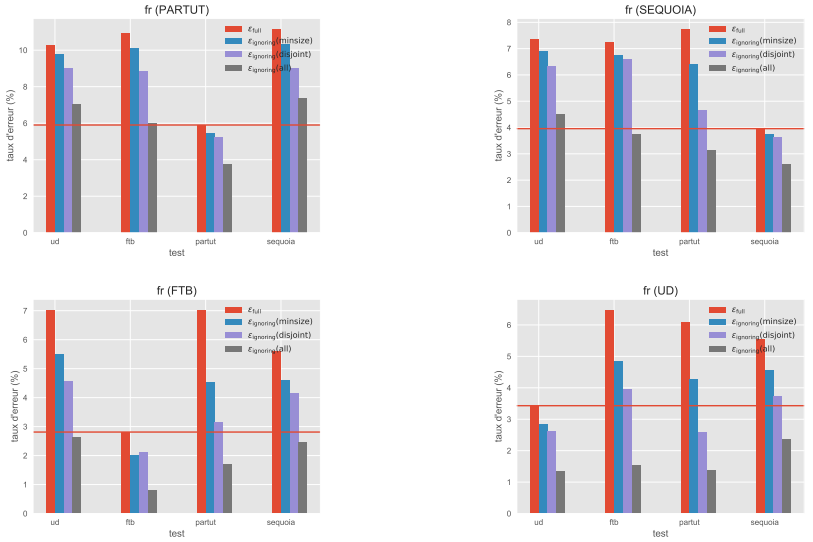


FIGURE 3: Taux d'erreur d'un analyseur morpho-syntaxique sur les différents corpus français de l'UD obtenus lorsque toutes les étiquettes des matchs sont correctement prédites. La ligne rouge représente le taux d'erreur obtenu sur un corpus de test du même domaine.

5 Conclusion

Dans ce travail, nous avons montré que, lors de l'évaluation d'un analyseur morpho-syntaxique sur des données hors-domaine, de nombreuses erreurs étaient dues à des divergence dans l'annotation des données. Une méthode permettant de quantifier cette divergence a également été décrite. Bien que nous n'ayons considéré que les corpus du projet UD et la tâche d'étiquetage morpho-syntaxique, la méthode décrite est très générique et peut être facilement applicable à d'autres corpus ou annotations (par exemple des annotations en dépendances), une tâche que nous aborderons dans nos futurs travaux.

Remerciements

Ces travaux ont été en partie financés par l'Agence Nationale de la Recherche (projet PARSITI, ANR-16-CE33-0021). Nous remercions les relecteurs pour leurs commentaires et suggestions.

Références

ABEILLÉ A., CLÉMENT L. & TOUSSENEL F. (2003). *Building a Treebank for French*, In A. ABEILLÉ, Ed., *Treebanks : Building and Using Parsed Corpora*, p. 165–187. Springer Netherlands : Dordrecht.

BARTENLIAN E., LACOUR M., LABEAU M., ALLAUZEN A., WISNIEWSKI G. & YVON F. (2017). Adaptation au domaine pour l'analyse morpho-syntaxique. In *TALN 2017 - 24e conférence sur le Traitement Automatique des Langues Naturelles*, Orléan, France.

BEN-DAVID S., BLITZER J., CRAMMER K., KULESZA A., PEREIRA F. & VAUGHAN J. W. (2010). A theory of learning from different domains. *Machine Learning*, **79**(1-2), 151–175.

BLACK E., JELINEK F., LAFFERTY J., MAGERMAN D. M., MERCER R. & ROUKOS S. (1992). Towards history-based grammars : Using richer models for probabilistic parsing. In *Proceedings of the Workshop on Speech and Natural Language*, HLT'91, p. 134–139, Stroudsburg, PA, USA : Association for Computational Linguistics.

BOSCO C., MONTEMAGNI S. & SIMI M. (2013). Converting italian treebanks : Towards an Italian stanford dependency treebank. In *Proceedings of the 7th Linguistic Annotation Workshop and Interoperability with Discourse*, p. 61–69, Sofia, Bulgaria : Association for Computational Linguistics.

BOUDIN F. & HERNANDEZ N. (2012). Détection et correction automatique d'erreurs d'annotation morpho-syntaxique du french treebank. In *TALN 2012 - 19e conférence sur le Traitement Automatique des Langues Naturelles*, p. 281–291, Grenoble, France.

BOYD A., DICKINSON M. & MEURERS W. D. (2008). On detecting errors in dependency treebanks. *Research on Language and Computation*, **6**(2), 113–137.

CANDITO M. & SEDDAH D. (2012). Le corpus Sequoia : annotation syntaxique et exploitation pour l'adaptation d'analyseur par pont lexical. In *TALN 2012 - 19e conférence sur le Traitement Automatique des Langues Naturelles*, Grenoble, France.

DICKINSON M. & MEURERS W. D. (2003). Detecting errors in part-of-speech annotation. In *Proceedings of the Tenth Conference on European Chapter of the Association for Computational Linguistics - Volume 1*, EACL '03, p. 107–114, Stroudsburg, PA, USA : Association for Computational Linguistics.

GUSFIELD D. (1997). *Algorithms on Strings, Trees, and Sequences : Computer Science and Computational Biology*. New York, NY, USA : Cambridge University Press.

LIPENKOVA J. & SOUČEK M. (2014). Converting Russian dependency treebank to Stanford typed dependencies representation. In *Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics, volume 2 : Short Papers*, p. 143–147, Gothenburg, Sweden : Association for Computational Linguistics.

NIVRE J., AGIĆ Ž., AHRENBERG L. & OTHER (2017). Universal dependencies 2.1. LINDAT/CLARIN digital library at the Institute of Formal and Applied Linguistics (ÚFAL), Faculty of Mathematics and Physics, Charles University.

PEDREGOSA F., VAROQUAUX G., GRAMFORT A., MICHEL V., THIRION B., GRISEL O., BLONDEL M., PRETTENHOFER P., WEISS R., DUBOURG V., VANDERPLAS J., PASSOS A., COURNAPEAU D., BRUCHER M., PERROT M. & DUCHESNAY E. (2011). Scikit-learn : Machine learning in Python. *Journal of Machine Learning Research*, **12**, 2825–2830.

PLANK B., JOHANNSEN A. & SØGAARD A. (2014). Importance weighting and unsupervised domain adaptation of POS taggers : a negative result. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, p. 968–973, Doha, Qatar : Association for Computational Linguistics.

SEDDAH D., SAGOT B., CANDITO M., MOUILLERON V. & COMBET V. (2012). The French Social Media Bank : a treebank of noisy user generated content. In *Proceedings of COLING 2012*, p. 2441–2458, Mumbai, India : The COLING 2012 Organizing Committee.

SHIMODAIRA H. (2000). Improving predictive inference under covariate shift by weighting the log-likelihood function. *Journal of Statistical Planning and Inference*, **90**(2), 227 – 244.

VILARES D. & GÓMEZ-RODRÍGUEZ C. (2017). A non-projective greedy dependency parser with bidirectional LSTMs. In *Proceedings of the CoNLL 2017 Shared Task : Multilingual Parsing from Raw Text to Universal Dependencies*, p. 152–162, Vancouver, Canada : Association for Computational Linguistics.

WISNIEWSKI G., PÉCHEUX N., GAHBICHE-BRAHAM S. & YVON F. (2014a). Cross-lingual part-of-speech tagging through ambiguous learning. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, p. 1779–1785, Doha, Qatar : Association for Computational Linguistics.

WISNIEWSKI G., PÉCHEUX N., KNYAZEVA E., ALLAUZEN A. & YVON F. (2014b). Apprentissage partiellement supervisé d'un étiqueteur morpho-syntaxique par transfert cross-lingue. In *Proceedings of TALN 2014 (Volume 1 : Long Papers)*, p. 173–183, Marseille, France : Association pour le Traitement Automatique des Langues.

ZHANG Y. & NIVRE J. (2011). Transition-based Dependency Parsing with Rich Non-local Features. In *Proceedings of ACL 2011, the 49th Annual Meeting of the Association for Computational Linguistics : Human Language Technologies*, p. 188–193, Portland, Oregon, USA : Association for Computational Linguistics.

Annotation en Actes de Dialogue pour les Conversations d'Assistance en Ligne

Robin Perrotin Alexis Nasr Jeremy Auguste
Aix Marseille Univ, Université de Toulon, CNRS, LIS, Marseille, France
prenom.nom@lis-lab.fr

RÉSUMÉ
Les conversations techniques en ligne sont un type de productions linguistiques qui par de nombreux aspects se démarquent des objets plus usuellement étudiés en traitement automatique des langues : il s'agit de dialogues écrits entre deux locuteurs qui servent de support à la résolution coopérative des problèmes des usagers. Nous proposons de décrire ici ces conversations par un étiquetage en actes de dialogue spécifiquement conçu pour les conversations en ligne. Différents systèmes de prédictions ont été évalués ainsi qu'une méthode permettant de s'abstraire des spécificités lexicales du corpus d'apprentissage.

ABSTRACT
Dialog Acts Annotations for Online Chats

Technical online chats distinguish themselves from more usual natural language production. They are written dialogs between two locutors that are meant to solve the user's problems. We describe here such chats with dialog acts annotation tailored for theirs specificities. Several prediction systems are described and evaluated, as well as a method that allows to learn models that abstract away from the lexical specificities of the training corpus.

MOTS-CLÉS : TAL, Traitement Automatique des Langues, Actes de Dialogues, Conversations en Ligne, Big Data, CRF, Réseaux Neuronaux.

KEYWORDS: NLP, Natural Language Processing, Dialog Acts, Online Chats, Big Data, Conditional Random Fields, Neural Networks.

Introduction

Ces dernières années ont vu le développement important de dispositifs de conversations instantanées en ligne, qui constituent un moyen pour de nombreuses entreprises et de services de fournir à leurs usagers un support technique. Les données générées par ces échanges constituent une trace linguistique d'une interaction entre un téléconseiller et un usager, visant à résoudre un problème rencontré par ce dernier.

Ces interactions sont intéressantes à plusieurs titres, d'un point de vue linguistique, elles permettent d'étudier sur des cas concrets la manière dont deux individus collaborent afin de résoudre un problème. Du point de vue du traitement automatique de la langues, elles constituent des productions écrites d'un genre original, souvent assez bruitées, qui donnent du fil à retordre aux outils de TAL. Du point de vue de l'entreprise, l'analyse de ces données pourrait permettre d'améliorer la qualité et l'efficacité

des échanges à la fois pour les entreprises et pour les usagers.

Nous proposons dans cet article d'annoter de telles données à l'aide d'actes de dialogues, décrivant la fonction illocutoire principale des tours dans le dialogue (Austin, 1975), notion linguistique qui a déjà été utilisée avec des jeux d'étiquettes variés pour le traitement du dialogue écrit (Core & Allen, 1997; Shriberg *et al.*, 2004; Ivanovic, 2005; Moldovan *et al.*, 2011; Salim *et al.*, 2016; Hardy *et al.*, 2003).

Une telle annotation permet de décrire le rôle que joue, dans le dialogue, chaque production des locuteurs. Elle peut participer à l'élaboration de tâches plus complexes, tel que le résumé automatique, la catégorisation automatique des conversations ou encore l'extraction d'informations. Notre étude porte plus spécifiquement sur des conversations en ligne provenant du projet ANR DATCHA [1]. Ces conversations proviennent des centres d'aide en ligne de l'entreprise Orange. Ces conversations sont de même nature que celles décrites par Damnati *et al.* (2016).

Nous avons développé un jeu d'étiquettes qui se concentre sur les aspects spécifiques des échanges où un usager et un téléconseiller coopèrent pour résoudre un enjeu dialogique [2]. Notre jeu d'étiquettes a été utilisé pour annoter un corpus, à partir duquel des outils d'annotation automatique ont été développés et évalués. Les données collectées dans le cadre du projet DATCHA, possèdent des caractéristiques générales à tout échange entre un usager et un téléconseiller ainsi que des caractéristiques spécifiques à un domaine d'activité donné, ici la téléphonie. Nous proposons d'essayer de nous abstraire du domaine d'activité en éliminant du lexique les mots spécifiques à ce dernier.

Cet article se décompose en trois parties. Il commence par une présentation de notre jeu d'étiquettes. S'ensuit une présentation de notre corpus de travail et de l'abstraction du lexique spécialisé en vue d'une généralisation des outils de prédiction. Dans un troisième temps, nous présentons les résultats obtenus à l'issue de la prédiction automatique de l'annotation en actes de dialogues.

1 Annotation en Actes de Dialogues

Le jeu d'étiquettes présenté ici a été conçu pour annoter des interactions langagières entre deux locuteurs distants lors d'une première interaction où l'un des locuteur (le Client C) présente un problème à l'interlocuteur (le Télé-Conseiller TC). Une spécificité importante de ces interactions est la collaboration des locuteurs en vue de la résolution d'un enjeu commun (le problème du client).

L'annotation décrit chacun des tours de parole (en respectant la segmentation des locuteurs) de la conversation par l'une des dix étiquettes présentées dans le tableau 1.

Cet ensemble d'étiquettes permet de décrire sommairement les tours, établissant leur fonction illocutoire principale [3] au sein du dialogue. L'annotation est volontairement simple et ne décrit pas tous les phénomènes dialogiques pouvant survenir. Un guide technique d'annotation (Asher *et al.*, 2017) décrit plus en détails ces étiquettes.

Dans l'exemple 1, un dialogue annoté extrait de notre corpus est présenté. Structurellement, le dialogue se construit par une courte séquence d'ouverture suivie de la description du problème à

1. http ://datcha.lif.univ-mrs.fr

2. Un enjeu dialogique étant une raison pour l'usager de contacter le téléconseiller, par exemple la résolution d'un problème ou une demande d'information.

3. Pour être plus précis, l'annotation permet le multi-étiquetage des tours, mais pour tout ce qui est présenté ici, seule la fonction principe a été conservée.

Étiquette	Signification	Description
OPE	Opening	Tours d'ouverture du dialogue
PRO	Problem Description	Description du problème du client
INQ	Information Question	Demande d'information de la part d'un des locuteurs
CLQ	Clarification Question	Demande de clarification
STA	Statement	Apport de nouvelles informations à l'interlocuteur
TMP	Temporisation	Mise en pause temporaire du dialogue
PPR	Plan Proposal	Proposition de résolution du problème.
ACK	Acknowledgment	Acquiescement des propos de l'interlocuteur
CLO	Closing	Tour de fermeture du dialogue
OTH	Other	Tour n'étant pas décrit par les autres étiquettes.

Tableau 1: Jeu d'étiquettes utilisé pour l'annotation en actes de dialogue.

résoudre (tour 2). Après plusieurs questions destinées à préciser divers aspects du problèmes (tours 4 et 7), le téléconseiller donne un début de réponse informel. Le tour 10 est une relance du problème de la part du client, finalement résolu au tour 12. La fin du dialogue est une suite d'échanges assez protocolaires visant à évaluer la qualité du service produit puis à clore le dialogue. L'annotation en actes permet de décrire toutes ces informations.

OPE	1	TC	Bonjour, je suis _TC1_, que puis-je pour vous ?
PRO	2	C	impossible pendant la lecture d'avancer la lecture
STA	3	C	_NUMTEL_
CLQ	4	TC	Si je comprends bien, le problème concerne la vidéo à la demande ?
STA	5	C	mais aussi l'**enregistreur** et la tv à la demande
INQ	6	C	pouvez vous m'appeler sur le portable ?
INQ	7	TC	Est ce que vous avez un message d'erreur ?
STA	8	C	non
STA	9	TC	Si vous avez **débuté** le **visionnage**, mais que le téléchargement n'est pas terminé, l'avance et le retour **rapides** sont **indisponibles**. Vous pouvez uniquement **stopper** ou reprendre le **visionnage** au début de votre vidéo.
PRO	10	C	seulement l'**enregistreur** avait terminé l'enregistrement et au cours de la lecture je n'arrive pas à avancer
CLQ	11	TC	Donc le téléchargement a terminé mais vous n'y **arrivez** pas à l'avancer ?
STA	12	C	après l'avoir **débranché** puis **rebrancher** ça **refonctionne** merci
INQ	13	TC	Ca fonctionne maintenant ?
STA	14	C	oui
ACK	15	TC	Parfait.
INQ	16	TC	Puis-je faire autre chose pour vous ?
STA	17	C	non merci
CLO	18	TC	Je vous en **prie** Mr _CLIENT_.
CLO	19	TC	Orange vous remercie de votre confiance. Je vous souhaite une bonne journée.

Exemple 1: Dialogue Annoté

2 Corpus DATCHA

Notre corpus de travail provient du projet ANR DATCHA. L'objectif du projet est de réaliser des tâches d'extraction d'information sur une grande base de données de conversations en ligne. Le corpus est constitué de conversations instantanées des différents services techniques et commerciaux d'Orange. Dans cette partie sont présentés d'une part les résultat statistiques de l'annotation manuelle partielle de ce corpus, et, d'autre part, la méthode que nous avons utilisée pour nous abstraire du corpus en retirant des informations relatives au lexique métier.

2.1 Annotation Manuelle du Corpus

Nous avons réalisé l'annotation manuelle de 2990 conversations dont la longueur moyenne est de 31,5 tours. Le tableau 2 montre la répartition des étiquettes dans le corpus. STA est l'étiquette la plus fréquente avec presque 40% des tours, tandis que TMP, CLQ et OTH représentent ensemble moins de 5% du corpus. Nous pouvons sommairement comparer la répartition de nos étiquettes avec d'autres travaux sur l'annotation des actes de dialogue, même s'il est difficile d'aligner parfaitement des jeux d'étiquettes différents.

Notamment, l'étiquette *Statement* existe et est utilisée à une échelle similaire dans Ivanovic (2005), Moldovan *et al.* (2011), avec respectivement 36%, 34.5% d'occurrences. Salim *et al.* (2016) a 57.1% de tours avec comme fonction Inform ou Answer.

En regroupant les différents types de questions (INQ et CLQ), notre corpus en contient 21% contre respectivement 19.2% pour Ivanovic, 10.6% pour Moldovan et 17.2% pour Salim pour les équivalents les plus directs dans leurs jeux d'annotation respectifs.

Les différences importantes s'expliquent par les différences de nature des corpus : les conversations de support clients étudiés par Ivanovic sont plus proches de notre corpus que les conversations spontanées de Moldovan ou les conversations de support entre clients (et non pas client/conseiller) pour Salim.

étiquette	occurrences	pourcentage
STA	36825	39.16%
INQ	18064	19.21%
PPR	14276	15.18%
ACK	6225	6.62%
PRO	5388	5.73%

étiquette	occurrences	pourcentage
CLO	5156	5.48%
OPE	3831	4.07%
TMP	2286	2.43%
CLQ	1638	1.74%
OTH	353	0.38%

Tableau 2: Statistiques par Étiquette

3027 tours de parole ont été annotés par deux annotateurs afin d'évaluer la précision du guide d'annotation et évaluer la tâche d'annotation. Avec un Kappa de Cohen de $\kappa = 0.67$, l'annotation manuelle peut paraître fortement dépendante de l'annotateur. Cependant, une analyse plus fine des divergences a montré qu'environ la moitié des divergences entre annotateurs proviennent d'omissions du guide, imprécis dans certaines situations fréquentes dans notre corpus. Toutefois de réelles ambiguïtés existent, notamment entre les deux types de questions (CLQ pour Clarification et INQ pour Information), ou encore pour distinguer Statement d'autres étiquettes, notamment PRO, PPR et ACK.

2.2 Abstraction du Lexique Spécifique

Notre corpus de travail n'est pas publiquement disponible, ce qui est souvent le cas pour ce type de données. Toutefois, l'annotation en actes de dialogue que nous proposons, ainsi que les modèles de prédiction appris sur nos données pourraient être utilisés sur une catégorie de corpus qui est bien plus large que les thématiques et spécificités lexicales propres à notre corpus. Pour essayer de s'abstraire de ces spécificités, nous avons tenté d'éliminer automatiquement du corpus la partie du lexique qui est trop liée à Orange ou aux tâches spécifiques à la résolution des pannes par son service technique. L'objectif est de produire des outils d'annotation automatiques utilisables dans pour des corpus différents.

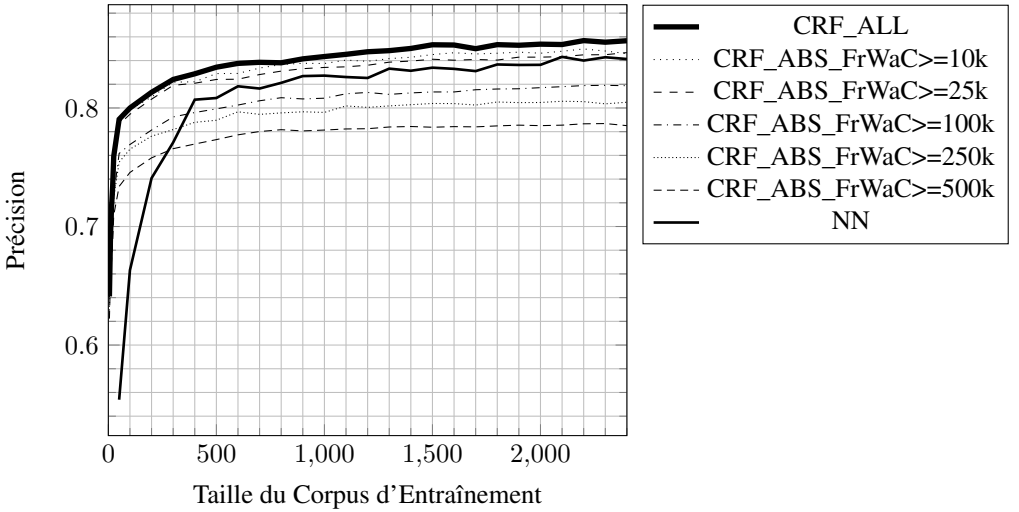
Pour construire ce lexique, nous avons extraits de frWaC (Baroni *et al.*, 2009), un grand corpus regroupant des productions de diverses sources sur le web, des lexiques de formes apparaissant plus d'un nombre fixé de fois. Par exemple, frWaC>=25k est le lexique des 5097 formes apparaissant plus de 25 000 fois dans frWaC. frWac>=500k est un lexique très restreint ne contenant que 284 formes différentes. À partir de chacun de ces lexiques, il est possible de filtrer les phrases de notre corpus de travail en considérant toutes les formes absentes du lexique comme inconnues. Dans l'exemple 1, les formes en gras sont celles absentes du lexique frWaC>=25k. Ce dernier conserve 45,5% des formes du lexique de notre corpus, soit 81,5% des occurrences produites par les locuteurs.

Parmi les 20 formes les plus fréquentes dans le corpus DATCHA qui ne sont pas dans frWaC>=25k et sont donc abstraites du corpus se trouvent 'livebox' 'décodeur' 'echat' 'sosh' 'box' 'wifi' 'câble' 'télécommande' 'chaînes' et 'redémarrer'. Ces mots sont de bons candidats pour l'abstraction. En revanche, des mots comme 'prie' 'confirmer' 'patienter' 'plait' ou 'svp' que l'on aurait souhaité conserver sont abstraits du corpus, n'apparaissant pas suffisamment dans frWaC. Il est difficile de mesurer à quel point cette méthode permettrait d'annoter des corpus proches dans la forme mais distincts dans les thématiques. Toutefois, nous présentons dans la section suivante des mesures expérimentales de l'impact de cet appauvrissement sur les performance de la prédiction automatique.

3 Prédiction des actes de dialogue

À partir de notre corpus annoté manuellement (respectivement 2390, 300 et 300 dialogues pour Train/Dev/Test), nous avons entraîné des modèles pour étiqueter automatiquement en actes de dialogue. Les résultats présentés dans cette section sont issus de trois modèles. [CRF_ALL], un modèle basé sur des Conditional Randm Fields (CRF) (Lafferty *et al.*, 2001) utilisant la totalité du lexique d'entraînement, [CRF_ABS_FrWaC>=N], une classe de modèles également basé sur des CRF mais dont le lexique est restreint par FrWaC, et [NN], un modèle basé sur des réseaux de neurones.

Les features utilisées pour les modèles CRF pour décrire un tour de parole sont : la longueur du tour, la position relative du tour dans le dialogue, quel locuteur a produit le tour et quels sont les mots apparaissant dans le tour (après restriction pour les CRF_ABS_FrWaC>=N). La longueur du tour est un ensemble de features binaires d'appartenance à une catégorie parmi dix, lesquelles ont été prédéterminées pour être équilibrées suivant le corpus Dev (ici les tours de 1 mots sont une catégorie, les tours de 2 mots une autre et les tours de 3 ou 4 mots sont regroupés ensembles, etc...). La position relative du tour suit la même idée de catégorisation mais pour des catégorie du type "le tour est dans le i-éme décile du dialogue".



Courbe 1: Courbe d'apprentissage des modèles suivant la quantité de données d'entraînement

Complexifier davantage le modèle n'a pas produit de meilleurs résultats dans nos tests. Cela est probablement dû à la nature de l'annotation, qui ne nécessite pas une analyse fine de la sémantique ou de la syntaxe des tours de parole. L'implémentation a été effectué par l'utilisation de CRFSuite (Okazaki, 2007). Pour catégoriser un tour par sa position relative dans le dialogue, l'algorithme sépare chaque dialogue en dix sections de taille égales et chaque tour est décrit par une unique feature précisant son segment. Cela a pour objectif de permettre à l'algorithme de pouvoir s'adapter à des conversations de taille très variable (d'une dizaine à quelques centaines de tours), tout en détaillant une information utile à l'analyseur (notamment, les tours de clôture sont souvent concentrés dans le dernier segment, peu importe la taille du dialogue).

Inspiré par Yang *et al.* (2016), NN est un réseau de neurones récurrent hiérarchique à deux niveaux. Le premier niveau s'intéresse aux tours de parole individuellement à partir de la séquence de mots de chaque tour, alors que le second niveau s'intéresse à la conversation dans son ensemble à partir des états cachés représentant les tours de paroles provenant du premier niveau. Les deux niveaux sont des réseaux récurrents bidirectionnels de type Long-Short Term Memory (LSTM). Pour un tour de parole i, la couche de décision utilise l'état caché i du dernier LSTM pour obtenir une prédiction de l'acte de dialogue du tour de parole. En entrée, le système utilise les mots, qui sont transformés par des embeddings de mots au premier niveau du LSTM, ainsi qu'une identification du locuteur.

La courbe d'apprentissage 1 présente l'évolution de la précision des classifieurs en fonction de la quantité de données d'apprentissage. Comme souvent, l'algorithme utilisant des réseaux de neurone a besoin d'une quantité plus importante de données pour arriver à des résultats comparables (ici au moins 500 conversations). Notre procédure d'abstraction du lexique affecte légèrement (de l'ordre de 1%) la précision du modèle, mais reste plus performante en tout point que NN pour FrWaC>=10k et FrWaC>=25k. Avec l'apprentissage sur tout le corpus d'entraînement, les résultats sont de 0.857, 0.847 et 0.841 pour respectivement CRF_ALL, CRF_ABS_FrWaC>=10k, et NN. CRF_ABS_FrWaC>=25k apparait comme un bon compromis entre taille du lexique et performances.

Une analyse des erreurs des meilleurs classifieurs nous a montré que l'étiquette la plus problématique,

est, sans surprise, l'étiquette STA. Sur notre corpus de test, 75% des erreurs d'annotation de nos meilleurs classifieurs sont en effet soit des annotations erronées de tours en STA ($\sim$45% des erreurs), soit annoté avec une autre étiquette alors que STA correspond à notre annotation manuelle ($\sim$30% des erreurs). Parmi ces erreurs, STA/PPR, STA/PRO STA/INQ sont les distinctions les plus difficiles pour nos modèles.

La ponctuation est souvent en cause pour les erreurs de type STA/INQ, en particulier lorsque le point d'interrogation est omis dans une question. Des tours comme *et que doit il se passer* ou encore *ok et le montant* sont incorrectement classifiés en STA, dans leurs contextes respectifs.

Pour STA/PRO, il est plus difficile de décerner une raison commune aux erreurs des classifieurs. Toutefois, il y a une fréquence plus élevé d'erreurs de classification au milieu des dialogues. Des tours comme *ça ne fait rien d autre que afficher perte de synchronisation* qui auraient été des PRO en début de dialogue sont au milieu de dialogue des STA, et les classifieurs étiquettent ceux-ci incorrectement.

Conclusion

Dans cet article, nous avons présenté et détaillé un jeu d'étiquettes pour décrire les actes de dialogues de conversations en ligne. Au-delà des résultats issus d'une annotation manuelle et de la classification automatique supervisée en utilisant soit des CRF soit des réseaux de neurones, nous avons proposé une méthode simple permettant de ne pas prendre en compte une partie du lexique spécialisé du corpus. Une poursuite naturelle de ces travaux est d'utiliser ces actes de dialogue dans une analyse de plus haut niveau, comme la catégorisation des conversations. Une autre piste de travail est l'élaboration d'une structure dialogique détaillant davantage l'interaction entre locuteurs, en s'appuyant sur les actes comme base.

Remerciements

Ces travaux ont été en partie réalisés grâce au soutien financier apporté par l'Agence Nationale pour la Recherche par le biais du projet DATCHA (ANR-15-CE23-0003).

Références

ASHER N., NASR A. & PERROTIN R. (2017). *Manuel d'annotation en actes de dialogue pour le corpus Datcha*. Rapport interne. `http://datcha.lif.univ-mrs.fr/files/dialogue-acts-manual-french.pdf` .

AUSTIN J. L. (1975). *How to do things with words*. Oxford university press.

BARONI M., BERNARDINI S., FERRARESI A. & ZANCHETTA E. (2009). The wacky wide web : a collection of very large linguistically processed web-crawled corpora. *Language resources and evaluation*, **43**(3), 209–226.

CORE M. G. & ALLEN J. (1997). Coding dialogs with the damsl annotation scheme. In *AAAI fall symposium on communicative action in humans and machines*, volume 56 : Boston, MA.

DAMNATI G., GUERRAZ A. & CHARLET D. (2016). Web chat conversations from contact centers : a descriptive study. In *LREC*.

HARDY H., BAKER K., BONNEAU-MAYNARD H., DEVILLERS L., ROSSET S. & STRZALKOWSKI T. (2003). Semantic and dialogic annotation for automated multilingual customer service. In *Eighth European Conference on Speech Communication and Technology*.

IVANOVIC E. (2005). Dialogue act tagging for instant messaging chat sessions. In *Proceedings of the ACL Student Research Workshop*, p. 79–84 : Association for Computational Linguistics.

LAFFERTY J., MCCALLUM A. & PEREIRA F. C. (2001). Conditional random fields : Probabilistic models for segmenting and labeling sequence data.

MOLDOVAN C., RUS V. & GRAESSER A. C. (2011). Automated speech act classification for online chat. *MAICS*, **710**, 23–29.

OKAZAKI N. (2007). Crfsuite : a fast implementation of conditional random fields.

SALIM S., HERNANDEZ N. & MORIN E. (2016). Comparaison d'approches de classification automatique des actes de dialogue dans un corpus de conversations écrites en ligne sur différentes modalités. In *23ème Conférence sur le Traitement Automatique des Langues Naturelles*.

SHRIBERG E., DHILLON R., BHAGAT S., ANG J. & CARVEY H. (2004). *The ICSI meeting recorder dialog act (MRDA) corpus*. Rapport interne, INTERNATIONAL COMPUTER SCIENCE INST BERKELEY CA.

YANG Z., YANG D., DYER C., HE X., SMOLA A. & HOVY E. (2016). Hierarchical attention networks for document classification. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics : Human Language Technologies*, p. 1480–1489.

Index

ATALA

ARIA

UMR IRISA

Inria
INVENTEURS DU MONDE NUMÉRIQUE

orange

UNIVERSITE BRETAGNE LOIRE MATHSTIC

METROPOLE Rennes
vivre en intelligence

INSA RENNES

Université Bretagne Sud
ubs

fondation RENNES 1

sylLabs

Script&Go

edp sciences

Qwant

Synapse